Welcome to the JReport Server User's Guide

This User's Guide describes JReport Server. JReport Server is a 100% Java report generation and management tool that can be deployed to any Java EE application server. By leveraging its high performance report generation engine, JReport scales to meet the most demanding requirements. Using the report scheduling, distributing and alerting capabilities of JReport Server, reporting can be integrated into the workflow of the application.

This guide is written for report administrators or system administrators of Java application that embed JReport reports into their application.

Other JReport documentation

This guide is one in the complete JReport documentation set. The documentation set includes the following:

- Getting Started with JReport
- JReport Tutorial
- JReport Server User's Guide
- JReport Designer User's Guide
- JReport Server Monitor User's Guide
- JReport API Javadoc

What's New in This Release

This document introduces the new features and significant enhancements of JReport in this release.

Embedding

Integration

- Embedding web reports, page reports and dashboards as a whole using JavaScript API for host applications in web browsers.
- Embedding web controls in host applications to open, save, export, filter, sort, print and navigate.
- API for applying and modifying NLS at runtime.

Connectivity

- RESTful web services as a data source with JSON formatted data with support for authentication and parameters.
- Unified semantic layer: Business View (BV) now integrates Report Cube (RC) and Business Cube (BC) features.
- User authentication when setting up notification e-mails.
- Backdoor admin authentication when bypassing LDAP is required.

Internationalization and Customization

- Full NLS support on product UI, data, metadata, contents, formulas, and resources.
- CSS enhancements with more objects and properties, as well as better category organization.

Other JReport Server Live Enhancements

- Custom aggregations using formulas.
- One-click sort and filter on table headers.
- Custom sort on table groups.
- Support for all built-in functions for crosstab.
- Exporting selected components from a report.
- Exporting multiple tabs of a page report into multiple sheets of an Excel file.
- Local parameter support on web reports and library components.
- Comparison function to compare crosstab values within reports and dashboards.
- External web controls via user defined dialogs.
- Configurable go-to-detail report window target (e.g., same window or new window).
- Default style group to Run, Advanced Run and Schedule reports.
- Setting a default report component when filtering using the Filter dialog.
- Dynamic Java home to start up/shut down Derby database when integrated in an application server.
- JRE only (without JDK) when integrated in an application server.
- Enable/disable CSRF (Cross-Site Request Forgery) checking for JReport Server Live.
- Support for synonyms when accessing Oracle databases.
- IE Navigate2 option for linking reports to URL when viewing in IE.
- Enhanced logs with more details.

UI and UX

Appearance

- Overall UI improvements including new styles and icons.
- Enhanced catalog data source view.
- Auto scaling Y-axis when scrolling through a Scrollable Chart.
- Formatting individual lines or a group of lines.
- Option in line charts to skip null data values in order to draw continuous lines.

Ease of Use

- Property Inspector to manage object properties in Web Report Studio.
- Windows Edge browser support.
- Direct linking from charts to reports without manually setting a URL.
- Enable/disable items in Options menu in JDashboard.
- Operate (e.g., sort, filter) on all fields of the dataset in a chart.
- New default for manually inputting filter values into Filter dialogs or optionally pulling filter values from the database.
- Select and operate on multiple items at a time in Business View Editor.
- Quick search in Data Mapping Editor.

Release Notes for Enhancements and Fixes

This document describes the feature enhancements, resolved issues and any known issues for JReport in this release.

Resolved issues

Item	Case #	Issue
1	66088	Resolved the XSS security issue when accessing a few JSP pages in JReport Sever.
2	66886	Resolved the ArrayIndexOutOfBoundsException error when loading reports in JDashboard.
3	67331	The parameter sessionid which is a given value in runReport.jsp can be displayed normally in the report's parameter page when calling UDS for the first time.
4	67331	JReport Server now calls the database only once when loading the parameter value list for bind column parameters which are bound to UDSs and retrieve data from the database in a web main report.
5	67331	The value list for the bind column parameters in a web main report will no longer be triggered when opening its linked report.
6	67914	Resolved the OutOfMemory issue when SSL is turned on with JReport Server.
7	68120	Resolved the jet.export.ExpException error when changing the width of an aggregation field in a page crosstab report to 0.01 and then exporting the crosstab to an Excel .xlsx file with column format.
8	68223	JReport no longer requests data from the data source when user tries to view through JReport Server the web report component whose Invisible property has been set to true.
9	68254	Solved the empty Excel result file problem when exporting a page banded report to Excel in JReport Server.
10	68255	Resolved the duplicated table and field names issue when merging two catalogs using Catalog Doctor in JReport Designer.
11	68278	JReport Designer can now create the node of hierarchy to business view normally when saving the catalog as XML.
12	68305	Resolved the IllegalArgumentException error when migrating JReport Server from version 5 to 13 by using command MigrationBV52.bat.
13	68331	The API function cat.getHandle(mapNames[i], CatalogAPI.COLUMN) now gets handled successfully when creating aggregations for an imported SQL using JReport Designer API.
14	68345	Resolved the java.lang.NumberFormatException error when adding a filter object to a page report that is running on Internet Explorer.
15	68365	Resolved the java.lang.ArrayIndexOutOfBoundsException error when scheduling a report using a bursting schema.
16	68365	The security error message can pop up properly in linked page reports.
17	68373	Corrected the 'Data type unmatched error' when doing format filter on the design mode of a web chart report and the field text field and the value text field of the condition are both filled with ToDate() expressions.
18	68390	The word wrap function now works normally on a page crosstab report when setting properties Suppress When Empty and Suppress When No Records to true.
19	68395	JReport Server no longer pops up the error page showing 'The report session has expired.' when user sets property server.session.timeout to 20 and performs continuous filters using the Filter dialog in Page Report Studio.
20	68422	Solved the GenerateFormulasCodeException error when saving a catalog after JReport Designer has been upgraded to V13. 1 Update 1.
21	68425	The resource permission on the Public Resource folder of JReport Server now gets enabled normally when using customized Security API.
22	68448	Support exporting page reports in IE 11 browser when Page Report Studio is embedded in a page via IFrame with HTTPS (SSL) URL.
23	68449	Corrected the error of a full user name in the Security > User panel of the JReport Administration page.
24	68550	The filter panel now fully displays in the edit mode of a web report.
25	68461	The scheduled tasks can get triggered normally when user schedules to run a few tasks at a specified time with option 'Run missed task upon server restart' checked, shuts down JReport Server before the specified time and then restarts it.
26	68540	Resolved the security issue of data access race condition that caused by storing user data in Servlet member fields.
27	68547	The page report formula on JReport Designer can now get correct value when calling isnull(next(@column)).
28	68563	Resolved the java.lang.ArrayIndexOutOfBoundsException: 0 >= 0 error when logging onto JReport Server in an LDAP configured environment.
29	68596	The new line at the bottom of the table column no longer disappears in the exported Excel workbook (.xlsx) of a page table report.
30	68597	Add a blank item to the From DB Catalog drop-down list in the Input SQL Name dialog when adding a new SQL to a data source so that database and schema name defined in the URL could be applied.
31	68601	The save as function can be performed normally in Page Report Studio when JReport Server is run with a JDashboard disabled license key.
32	68601	The formatting changes of a page report no longer gets lost after saving it to a report folder to overwrite the report with the same name.
33	68608	JReport Server now pops up with a clearer error message when someone exports a web report with invalid file name characters in the Export dialog.
34	68667	Resolved the ClassCastException error when selecting Top N, specifying "other" group and using category sort in Category Options dialog of a web report.
35	68679	The node icons on the legend of a page chart report now display normally.
36	68684	Labels in web reports can be edited successfully now when their width extends beyond the page in JReport Server.
37	68688	The field items available for the Conditional Formatting dialog in a web crosstab report are staying the same after JReport Server is upgraded to a higher version.
38	68696	Solved the payment_type_name field value missing issue on Page Report Studio when Group By Transaction Type option of parameter p_GroupByTransactionType gets unchecked in the Enter Parameter Values dialog.
39	68726	Solved the ClassCastException and ConcurrentModificationException errors when initializing multiple threads concurrently to call the method runReport.
40	68728	Resolved the NullPointerException error when creating cached query result for several queries in JReport Designer.
41	68746	The Edit Business View Security dialog appears normal when right-clicking Business View Security for a catalog and then select Edit Business View Security from the shortcut menu.
42	68748	Resolved the script error when opening a report folder and its subfolders in the server resource tree in Icons or Thumbnails view.
43	68798	Resolved the data type unmatched error when choosing to use a dynamic formula in a web report format filter.
44	68816	Resolved the java.lang.NullPointerException error when running a page report in JReport Server with trace level logs set to TRIVIAL.
45	68835	Lines in page reports are now displayed normally when viewing the reports in JReport Server.
46	68842	Resolved the jave.lang.NullPointerException error when doing a drill down action in a library component running in JDashboard.
47	68842	Web Report Studio now shows the field's new dynamic display name if a user modifies the dynamic display name of a field and adds the field to an existing ad hoc web report.
48	68863	JReport Server now shows a loading icon when a new filter is inserted in the Filter panel of Web Report Studio.
49	68880	User can now successfully insert new imported SQL files to a catalog using JReport Catalog API.
50	68880	The column elements under the SQL element now get updated successfully after an SQL file is updated via JReport Catalog API.
51	68925	Enhanced JReport Server performance when flex key CPU licensing is used.
52	68940	Fixed the unknown exception error when linking to a report in Page Report Studio.
53	68964	The summary result is displayed correctly in JReport Designer when creating a library crosstab component that uses a dynamic formula as aggregate field.
54	68964	The dynamic formula which is being used as an aggregate field in a library component of a crosstab now works normally with conditional format when the library component is being viewed on JReport Designer or run in JDashboard.
55	68969	Resolved the 100% CPU utilization problem when there is some typo on the keystore file name that user provides to enable SSL on JReport Server.
56	69054	The page crosstab report which uses table style and has two or more repeated fields no longer is displayed with a part of the field values or additional columns when previewed in JReport Designer.
57	69065	The page header panel which is set to be repeated on each page of a page report can now be displayed correctly on each sheet of the report's exported Excel result.
58	69078	Updated the file welcome.jsp of JReport Server for a typo error.
59	69082	The label of the field added to a web report is now displayed with its dynamic display name by default in Web Report Studio if user has configured a dynamic display name for the field.
60	69082	Resolved the java.lang.reflect.InvocationTargetException error when adding parameter control to a web report in JReport Designer.
61	69082	The standalone JReport Server running on Windows 7 on Internet Explorer 11 no longer gets two headers after the submission of a schedule task.
62	69090	The connection name of an imported SQL file can now get changed accordingly on JReport Designer after changing the name of the same catalog's JDBC connection via Catalog Doctor.
63	69106	Setting property Keep Group Together of a table group in a page report no longer results in a blank report page when running the report in Page Report Studio.
64	69131	Allow users to add new categories or hierarchies to their business views in any catalog on JReport Designer.
65	69133	Resolved the scrollbar and chart tooltips displaying issue when running JDashboard with multiple dashboard tabs via URL.
66	69160	The Query Filter dialog in a web report no longer shows an empty value list if a user has added a filter to this query on JReport Designer and the filter condition is on parameter.
67	69219	Enhanced JReport Server performance when running web reports with conditional formatting.
68	69219	Resolved the NullPointerException error when opening a linked report in another linked report in Web Report Studio.
69	69219	Improved JReport Server performance when opening linked reports in a web report created using user data source (UDS).
70	69219	Improved JReport Server performance when running a web report created with user data source (UDS) and the cached report data in the report is expired.
71	69219	Web report charts can be loaded normally in Web Report Studio which is running in Internet Explorer in case filters has been applied or removed in this report.
72	69219	Proper working of undo actions when two or more sorts have been applied to a web table report in JReport Server.
73	69221	The access privileges error message now pops up normally for web reports running in Google Chrome.
74	69249	Resolved the java.lang.ClassCastException error when calling method catalog.delete (sqlHandle) of Catalog API instance to delete an imported SQL object from a catalog.
75	69273	Dynamic formulas that are created as aggregation can now be used to create summary tables on JReport Designer plus crosstabs which use these dynamic formulas can show aggregation values normally in Web Report Studio.
76	69288	The error pages in JReport Server can now be closed with the Close button.
77	69300	The new line character '\r\n' added in the formula now works normally in the web table report running in Web Report Studio.
78	69365	Table alias no longer gets changed in the where condition of the SQL statement when adding a table in the catalog using JReport Catalog API.
79	69367	Resolved the java.lang.NullPointerException error when visiting JReport Administration page with a JReport Server running on Tomcat 8.
80	69411	Resolved the NullPointerException error when running a page report on a JReport production server.
81	69417	Resolved the NullPointerException error when creating a web crosstab report with the property Push Down Group Query of the JDBC connection being set to true in JReport Designer.
82	69461	The last row (i.e. total/aggregation) of a page crosstab report now gets displayed properly when it shows up in the next page.
83	69467	The schedule task of publishing a report to e-mail now can be submitted successfully using Internet Explorer 11.
84	69472	The content of the error message in the unknown exception for Query Optimizer can be displayed normally in linked reports.
85	69493	Resolved the OutOfMemory issue when sorting fields and group by fields in a page report that has a large amount of data.
86	69513	The filter control of a web report no longer is exported to Excel after user sets the Export to Excel property of the web report filter control to false on JReport Designer.
87	69556	Web reports with dynamic formulas running in Web Report Studio now show consistent result with the same reports in JReport Designer.
88	69593	The error message now shows consistent date format with the user defined one when user enters an invalid date format in JReport User's Enter Parameter Values dialog.
89	69621	The description of binded column in the page header of a page banded report now stays as it is when JReport Designer is upgraded from V12 to V13.
90	69650	Enables user to insert tables to a catalog using method insert(java.lang.String dataSourceName, java.lang.String tableName, java.lang.String catalogName, java.lang. String schemaPattern, java.lang.String tablePattern, int type, java.lang.String character) when the method is simultaneously used on database catalogs and schemas.
91	69678	The suppress property now works normally in a page report in which a formula with function IsNoRecord() is used to control this property.
92	69684	The web report label whose width extends beyond the report page in JReport Server no longer disappears when it is no selected.
93	69705	Creating web reports using the wizard that has been embedded into an IFrame is now smoother.
94	69708	User can no longer run an inactive report via URL or JReport Server API.
95	69729	Rebuilds the installation file of JReport Server 13.1 Update 2 to ensure its successful installation in the console interface.
96	69735	Users now have no problem connecting to MongoDB 3.0 and loading tables on JReport Designer once they have a user role with the privilege of Read.
97	69737	The search function can now work normally in a parameter control in Page Report Studio.
98	69739	JReport Server now displays the business view tree correctly in the Group screen when creating an ad hoc page report with a business view and the first category in this business view has no sub node.
99	69755	Resolved the CSS parsing javascript error in Web Report Studio by preventing business view elements' names from starting with numbers.
100	69769	The parameter value's drop-down control no longer persists when user tries to enter a parameter value manually in the Enter Parameter Values dialog and press the Tab key to move onto the next parameter.
101	69807	Keep the Web Report Wizard window sizable when adding a table component to a web report.
102	69869	Support logarithmic scaling on the Y-axis of all chart objects running in Page Report Studio, Web Report Studio and JDashboard.
103	69892	The security setting for non-admin users now works normally in JReport Server when method getFolderInPath is called in the resource manager.
104	69892	Upgrading JReport Server from V11 Update 1 to V13.1 Update 2 now makes the horizontally aligned icons in Page Report Studio consistent.
105	69892	Fixed the error information getting lost in the Error.log of JReport Server 13.1 Update 2.
106	69894	Resolved the merged cells and extra columns in the Excel result of a page crosstab report when the report is exported to Excel with column format and the width of a certain column in the crosstab has been changed.
107	69938	The web controls are now kept in right position in Page Report Studio when JReport Server is upgraded from V11 Update 1 to V13.
108	69941	Resolved the access denied error when hitting OK button in a web report Save As dialog or hitting Save button in the Bind Data screen of the Web Report Wizard before the contents in both dialogs are loaded.
109	69958	The Reset button in the Enter Parameter Values dialog can now function normally after JReport Server is upgraded from V11 to V13.
110	69997	Solved the OutOfMemory issue caused by the heavy load on a specific JReport cluster node.
111	69999	Support default display sequence of parameters in the parameter form control for cascading parameters in a web report.
112	70037	The date values in a web report now get displayed correctly when the report is previewed or exported in Excel format.
113	70112	Support running JReport Designer and JReport Server using Sun's Java Runtime Environment (JRE) in user's production environment.
114	70127	JReport Server 10 Update 1 now works normally when running on JDK 1.8.
115	70141	The filter control now works successfully for a table which locates in the page header panel of a page report.
116	70156	Resolved the IndexOutOfBoundsException error when running a dashboard component with cached cube.
117	70237	Resolved the NullPointerException error when passing a parameter to a linked report in Web Report Studio.
118	70262	Removed the invalid object name error when adding a table to a catalog in the JDBC connection without specifying a table schema.
119	70312	User now has no problem selecting a folder in the New Schedule dialog with Internet Explorer 11 when accessing JReport Server with a domain name.
120	70335	Recovered the word wrap function for label texts in page crosstab reports.
121	70353	Table detail rows in a library component are displayed correctly when running JDashboard against cube.
122	70369	Improved JReport logging so additional errors and exceptions can be logged when issues occur.
123	70421	Removes the vulnerability in JReport products with updated lib files containing serializer-2.7.2 in makewar.xml file.
124	70441	Removes the java.lang.NullPointerException error when running a bursting report on JReport Server.
125	70451	The version date sorting now works properly in the Download from JReport Server dialog.
126	70492	The query that contains the special field "UserName" can no longer be used to create in-memory cube.
127	70531	Fixed the reports hanging issue when running reports on JReport Server.
128	70552	Removes the javascript error when browsing the JReport Server resource tree or selecting a folder via Internet Explorer with its compatibility version being set to 7.
129	70565	Labels of the scrollbar no longer are overlapped when running a library component of a scrollbar chart on JDashboard.

Known issues

Summary field disappears from design view when deleting table column

If a summary is inserted in the GroupFooterPanel of a table, when you delete a column from the table, or unmerge the cell of the group footer row, sometimes you may find the summary field disappears from design view, however, from the report structure tree in the Report Inspector, you can still find the object. You can reset its coordinate related properties so as to make the summary field display again.

Compiling formulas in JReport under JDK 6 or higher gets warnings

Since generic type is introduced to JDK 6 or higher version, when compiling formulas in JReport under JDK 6 or higher, you may come across warning messages as follows:

- Note: Test.java uses unchecked or unsafe operations.
- Note: Recompile with -Xlint:unchecked for details.

You may see that the situations are logged as Javac errors. However, the warning messages do not prevent Javac from creating .class files and reports can still run correctly.

In this case, you can simply ignore the messages and logged errors.

TOC Browser tree doesn't work well for reports with cached report bursting

When end user runs a report which has been defined with some cached report bursting policy in Page Report Studio, and then opens the TOC Browser, he will see all the groups of the report instead of just the ones he is supposed to view. In addition, no matter if he has access to a group, he cannot click the group name from the TOC tree to get corresponding details. JReport will further enhance the cached report bursting feature in a future release to resolve this issue.

Report data gets cut off in PDF result

When you export a report to PDF format, if the report contains a large amount of data but its page mode was specified to be continuous page mode, or its page size was set to be larger than 200 inches, you will find that in the PDF result some data of the report are cut off. This is because in one PDF page, the data displayed can be no larger than 200 inches.

Starting JReport with 64-bit JDK 1.7

To use 64-bit JDK 1.7 to start JReport on 64-bit Sun Solaris System, you need to modify the launch script JReport.sh (for JReport Designer) or JRServer.sh (for JReport Server) in `$REPORTHOME/bin` by changing the value $JAVAHOME/bin/java to $JAVAHOME/bin/amd64/java.

Deploying monitor.war (servlet.war) and jreport.war in the same domain gets exception

When integrating JReport Server and JReport Server Monitor into one application server, you need to make sure that they are deployed to different domains. JReport Server Monitor is supposed to be just an application for the administrator which doesn't need to come along with a JReport Server as the report system. So it is recommended that JReport Server Monitor be installed on a separate system or on systems for system administrators.

Limitation of going to another group on chart

When you perform the go-to action on a chart which uses a dynamic formula as its shown value, and the formula contains group information, you will get exceptions. This is a limitation in current version. We will resolve the issue in future release.

The zooming in or out of JReport browsers is not supported

All JReport browsers such as Page Report Studio, Web Report Studio, JDashboard and so on do not support being zoomed in or out by the way of scrolling the mouse while holding the Ctrl key. This may result in that the JReport UI cannot display well.

API change

jet.formula.ParamDesc.value has been set to private now. You can use getDisplayValue(DbValue paramvalue, java.util.Locale locale) instead.

For example:

```
ParamDesc desc = new ParamDesc();

... ...

desc.getDisplayValue(desc.getValue(), Locale.getDefault());
```

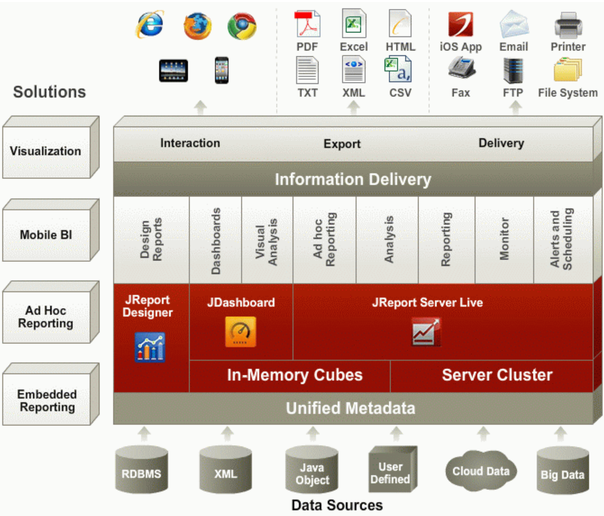
JReport Product Overview

JReport delivers operational business intelligence to enterprise applications through powerful embedded reporting.

JReport is a complete Java reporting solution that provides sophisticated enterprise reporting, ad hoc reporting, and data analysis. A 100% Java EE architecture and a rich set of APIs allow JReport to be seamlessly embedded into any application, providing end users with a transparent interface to easily generate reports, share information, and analyze data. With JReport, any report can be made interactive, extending the "life" of a report by allowing users to easily sort, group, navigate, and filter via the Web. This wide range of functionality, including the ability to drill down on data, enables users to quickly derive value from their business intelligence.

JReport product architecture

JReport's Java architecture takes advantage of the portability, scalability, and ease of integration associated with J2EE technology to provide a powerful, flexible reporting solution that fits perfectly within any architecture.



JReport Designer is a Swing-based Integrated Development Environment (IDE) that enables sophisticated report design and presentation of critical business data. It provides an intuitive interface, reusable report components, flexible layout, and a toolset for designing and testing reports. With JReport Designer, you can build reports using simple drag and drop techniques or by using the Report Wizard. Data can be accessed from any data source to design and preview reports in order to deliver information to end users in the most relevant and intuitive manner. Rapid creation and modification of reports is accomplished by toggling between design mode and view mode where the report will be displayed with the actual dataset. Once report design is complete, the report is published to JReport Server for generation, delivery, and management.

JReport Server is a 100% Java report generation and management tool. It enables efficient management, sharing, scheduling, versioning, and delivery of reports and enables reporting to be integrated into the workflow of any Java application. The high-performance engine can scale to any workload. Report results can be saved to a versioning system, sent to enterprise/workgroup printers, or e-mailed. With JReport, reports can be viewed in any modern enterprise format including Page Report, Web Report, HTML and standard business documents, such as PDF, Excel, and RTF.

JDashboard delivers information using a user portal user interface rather than a report. Users can freely choose the objects they want to display in the dashboard, without having to know how these objects were created, what data sources to use, what styles to set, and so on. A dashboard can hold multiple data components so that when browsing the dashboard users are able to see multiple data aspects. Within a dashboard, data components are able to communicate with each other via the message mechanism. This allows actions such as common filters to be applied to all the components of a dashboard even when coming from different data sources.

Page Report Studio and **Web Report Studio** enable reports to be accessed through a web browser via Dynamic HTML, or AJAX. With Page Report Studio and Web Report Studio, reports can be modified using dynamic filter, sort, and drill capabilities. Using Page Report Studio and Web Report Studio's advanced capabilities, users can drag and drop columns to and from an existing report, dynamically change chart types, pivot crosstabs, add groups, convert report components or create an entirely new report.

Visual Analysis is a WYSIWYG product to visualize the result of every work step. Simply by dragging and dropping data fields onto a layout module, users are able to visually create crosstabs and charts step by step. The use of colors, sizes, shapes, and pie slices demonstrates the data in rich aspects.

Reporting a problem or requesting a feature

If you are having trouble running JReport Server or encounter any problems during reporting, take the following steps.

1. Check whether the system on which JReport Server is running meets the system requirements.
2. Check the JReport FAQ pages for frequently-asked questions and their solutions.

If the problem persists, report it to Jinfonet Support (support@jinfonet.com) with the following information:

1. Describe the precise steps leading to the problem.
2. Run the batch file jrenv.bat in the `<install_root>\bin` subdirectory. Running this batch file will generate a file called report.env in the current directory. Send this file to Jinfonet Support. Also, describe the operating environment, including machine type, CPU, memory, OS, and Java version.
3. If you are running JReport in an integrated Java application server, click the **Server Information** button on the JReport Administration page to list the environment.
4. Send Jinfonet Support the log file with the recorded JReport Server problems.
   - **For a standalone server**
     Start JReport with the batch file DJRServer.bat (.sh on Unix). Running this batch file will record the most detailed logging information and write them to the log files in the `<install_root>\logs` directory. Try to reproduce the problem, and send the log files along with the other information.
   - **For an integrated JReport Server**
     If your JReport Server runs as a servlet inside a Java application server, send the log files generated by the JReport Engine in the application server. When the JReport Servlet is installed, a property file is generated which is used to define the class and arguments of the JRServlet. For example, if you integrated JReport Server into WebLogic, find the WebLogic properties file located in the installation path of WebLogic. Edit the file to use the option -vDebug -vError. Your file should then contain the following content:

     ```
     vError=true
     vDebug=true
     ```

     In addition, you can also directly add the java option -Dlogall=true (or -DvError=true, -DvDebug=true) to the java command line within the launch file of the application server.

     After restarting your application server, reproduce the problem and send the log files to Support.
5. Send Support the log files recording all of the logging information of engine and server (including event, error, debugging, access, management, and performance).
   - Change the configuration to record all logging events by starting the server and accessing the JReport Administration page through `http://localhost:8889`.
     a. Go to the **Configuration** > **Log** panel.
     b. Set the trace level of all logs to **TRIVIAL**, and set the error level of all logs to **WARN**.
     c. After that, reproduce your problem and send Support the log files in `<install_root>\logs`. In addition, you can also directly modify the logging configuration file LogConfig.properties in `<install_root>\bin`. If you set the server property log.config.update to true (all server properties are managed within the file server.properties in `<install_root>\bin`), any changes to the configuration file will automatically take effect at runtime after the specified update interval (set by the server property log.config.update.interval).
   - If your JReport Server runs as a servlet within a Java application server, in the address bar of a web browser, type in `http://<hostname>/jreport/admin`, where jreport is the servlet context path.
6. To reproduce your problem of running reports with JReport Server, we will often need your report, catalog and data information.
   a. Send Support the catalog file (*.cat and *.fml) and the report file *.cls that you are having problems with.
   b. In order to resolve technical issues that you have reported, we will need to access your report data so that we can recreate and analyze the problem. Your database may be very large. However, we will only require access to the data returned by the query of the troublesome report, and if necessary we will sign a confidentiality agreement with you. To extract the report data, in the Catalog Browser of JReport Designer, right-click the query that your report is using, select the menu item **Create Cached Query Result**. Then, input the data file name and click the **Save** button. The query result will then be saved in this file. Send Support all of the files generated (including the description file).

Visiting web pages for more information

- Products information
  http://www.jinfonet.com/products
- News center
  http://www.jinfonet.com/company/news
- Demo center
  http://www.jinfonet.com/resources/jreport-demo
- Products download center
  http://www.jinfonet.com/product/download-jreport
- White papers
  http://www.jinfonet.com/resources/white-papers
- JReport product documentation
  http://www.jinfonet.com/jreport-documentation
- JReport Javadoc
  http://www.jinfonet.com/jreport-documentation
- JReport Technical Support center
  http://www.jinfonet.com/services/technical-support

Installing and Uninstalling JReport Server

This chapter presents how to install JReport Server using a variety of different methods. It covers issues associated with installing, removing and solving problems encountered during installation.

- System requirements
- JReport licenses
- Supported report databases
- Installing using the Installation Wizard
- Installing silently
- Installing using the console interface
- Installing on Unix manually
- Installing by building a war file on Windows to deploy to Linux/Unix
- Uninstalling JReport Server
- Solving installation problems
- JReport Server reporthome directory overview

System requirements

The following table displays the basic system requirements for installing JReport Server. Check your system to make sure that all the requirements are met before installation.

JReport Server System Requirements		
	Recommended Requirements	**Minimum Requirements**
OS:	Windows x64, Unix x64, Linux x64, z/Linux64	Windows, Unix, Linux, z/Linux
CPU:	Quad Core processor	Dual Core processor
Free Memory:	8 GB	2 GB
Free Disk:	10 GB	1 GB
JDK:	6 or above	6 or above
Browser:	Latest releases of Chrome, Firefox, IE, Edge	Chrome 23, Firefox 20, IE 9, Edge

Reference: Download a JDK version at http://www.oracle.com/technetwork/java/javase/downloads/index.html.

Notes:

- Jinfonet supports Java VMs released by Sun and IBM to run with JReport Server. You can try using other Java VMs, but their compatibility cannot be guaranteed. Reports of any problems you find with other Java VMs are welcome.
- If you want to use JReport Server on a z/Linux system, you must download the JDK specially used for IBM from http://www-03.ibm.com/servers/eserver/zseries/software/java/ and the version should be at least V6.
- You are not recommended to run JReport in the Internet Explorer Compatibility View mode.

JReport licenses

JReport has several add-on licenses which enable specific features:

- JReport Live license
- JReport Server Cluster license
- JDashboard license
- Visual Analysis license
- Organization license

Contact your Jinfonet Software account manager to obtain your required license.

JReport Live license

The JReport Live license enables the use of web reports and ad hoc page reports and all their related functions. JReport Designer and JReport Server have separate live licenses.

Live license for JReport Designer

A JReport Live license for JReport Designer controls the data sources for ad hoc page reports and web reports and the creation of web reports from those data sources on JReport Designer. The Live license allows you to utilize the following features on JReport Designer:

- Create and edit business views. They are meta-data descriptions created in JReport Designer to be used as data sources for ad hoc page reports and web reports. They can be created on any type of data source such as JDBC, ODBC, XML and Web Services.
- Create reports from business views.
- Export and print web reports.
- The following functions also require that JReport Server has a Live license:
  - Preview web reports in Web Report Studio.
  - Publish web reports to JReport Server.
  - Download reports which use business views as the data source from JReport Server.

Live license for JReport Server

A JReport Live license for JReport Server controls the real time ad hoc and analysis reporting on JReport Server. The Live license allows you to utilize all of the following features and functions on JReport Server:

- Create, view, run, edit, export, print, and publish ad hoc page reports and web reports. The data sources for these reports are created in JReport Designer, which requires JReport Designer has a Live license.
- All actions in Page Report Studio involving business view or changes of report template:
  - Create new page reports or page report tabs
  - Delete report tabs from a page report
  - Add components and data fields into reports
  - Remove components from reports
  - Move and resize components
  - Edit component properties
  - Drill
  - Change chart type
  - Rotate tables and crosstabs
  - Convert between crosstab and chart
  - Create query filters
  - Display the Resource View panel which shows the data resources for the current open report tab

JReport Server Cluster license

JReport Server Cluster license enables a group of JReport Servers to work together with shared resources, load balancing, and failover in a distributed cluster.

JDashboard license

A JDashboard license enables the use of JDashboard and all related functions. JReport Designer and JReport Server have separate JDashboard licenses.

Since JDashboard requires using business views as data sources, the JReport Live license is also required. JReport Designer and Server have separate Live licenses too.

JDashboard license for JReport Designer

A JDashboard license for JReport Designer allows for creation of library components which are used to build dashboards and publishing of library components to JReport Server.

JDashboard license for JReport Server

A JDashboard license for JReport Server enables the management of library components in the component library, the creation of dashboards using library components, and the use of dashboards.

Visual Analysis license

A Visual Analysis license enables the use of Visual Analysis and all related functions. Since Visual Analysis requires using business view as the data source, the JReport Live license is also needed in order to perform visual analysis.

Organization license

An organization license enables organizing users into different groups with their own administrators.

Supported report databases

JReport supports all of the current mainstream databases as well as most databases which support ODBC or JDBC drivers. Except for traditional databases, databases on the cloud are also supported, such as Vertica, Amazon RDS, and RedShift.

The following table lists the databases and JDBC drivers that have been tested with JReport. If you are using any of the databases listed below, you are recommended to use the corresponding driver version with JReport although any driver which the DBMS supplier recommends is also fine. If you encounter problems when using a database or driver version that is not listed here, you can contact Jinfonet Support (support@jinfonet.com) for help.

You can also refer to the page http://wiki.netbeans.org/DatabasesAndDrivers for additional information on database and driver.

Database	Version	Driver File Name	JDBC Driver	Example URL
MS SQL Server	2014 (12.00.2000)	sqljdbc4.jar	com.microsoft.sqlserver.jdbc.SQLServerDriver	jdbc:sqlserver://<host>:1433;DatabaseName=test
MS SQL Server	2012 (11.00.2100)	sqljdbc4.jar	com.microsoft.sqlserver.jdbc.SQLServerDriver	jdbc:sqlserver://<host>:1433;DatabaseName=test
MS SQL Server	2008R2 (10.50.4000)	sqljdbc4.jar	com.microsoft.sqlserver.jdbc.SQLServerDriver	jdbc:sqlserver://<host>:1433;DatabaseName=test
MS SQL Server	2008R2 (10.50.4000)	Merlia.jar;	com.inet.tds.TdsDriver	jdbc:inetdae7:<host>:1433?database=test
MS SQL Server	2008	sqljdbc4.jar	com.microsoft.sqlserver.jdbc.SQLServerDriver	jdbc:sqlserver://<host>:1433;DatabaseName=test
MS SQL Server	2005	sqljdbc.jar	com.microsoft.sqlserver.jdbc.SQLServerDriver	jdbc:sqlserver://<host>:1433;user=sa;password=1234;database=test
MS SQL Server	2000	msbase.jar; msutil.jar; mssqlserver.jar	com.microsoft.jdbc.sqlserver.SQLServerDriver	jdbc:microsoft:sqlserver://<host>:1433
MS SQL Server	2000	Opta2000.jar	com.inet.tds.TdsDriver	jdbc:inetdae7:<host>:1433?database=test
MS SQL Server	2000	tds-1.0.3.jar	net.sourceforge.jtds.jdbc.Driver	jdbc:jtds:sqlserver://<host>:1433/test
MemSQL	5.5.8	mysql-connector-java-5.1.25-bin.jar; mysql-connector-java-5.1.7-bin.jar	com.mysql.jdbc.Driver	jdbc:mysql://<host>:3306/test
ScaleDB	0.2.3	mysql-connector-java-5.1.25-bin.jar; mysql-connector-java-5.1.7-bin.jar	com.mysql.jdbc.Driver	jdbc:mysql://<host>:3306/test
MySQL	5.5.24 (64bit)	mysql-connector-java-5.1.25-bin.jar; mysql-connector-java-5.1.7-bin.jar	com.mysql.jdbc.Driver	jdbc:mysql://<host>:3306/test
MySql	mysql via SSL	mysql-connector-java-5.1.6-bin.jar	com.mysql.jdbc.Driver	jdbc:mysql://db06:3306/test?p?useSSL=true?clientCertificateKeyStoreUrl= D:\test\SSL_Client\ca-cert.pem?clientCertificateKeyStorePassword=1234
MySql	mysql 5	mysql-connector-java-5.0.4-bin.jar	com.mysql.jdbc.Driver	jdbc:mysql://<host>:3306/test
MySql	mysql-5.0.2-alpha-win	mysql-connector-java-3.1.5-gamma-bin.jar	com.mysql.jdbc.Driver	jdbc:mysql://<host>:3306/test
MySql	mysql-5.0.18-win32	mysql-connector-java-5.0.3-bin.jar	com.mysql.jdbc.Driver	jdbc:mysql://<host>:3306/test
MySql	mysql-4.1.12-win32	mysql-connector-java-3.1.10-bin.jar	com.mysql.jdbc.Driver	jdbc:mysql://<host>:3306/test
MySql	mysql 4	mysql-connector-java-3.0.14-production-bin.jar	com.mysql.jdbc.Driver	jdbc:mysql://<host>:3306/test
Oracle	11.2.0.1.0 (64bit)	classes12.jar; ojdbc14.jar; ojdbc5.jar; ojdbc6.jar	oracle.jdbc.driver.OracleDriver	jdbc:oracle:thin:@<host>:1521:orcl (Oracle JDBC Thin using an SID)
Oracle	11.1.0.6.0 (11g)	JDK1.5: ojdbc5.jar; JDK1.6: ojdbc6.jar; classes12.jar	oracle.jdbc.OracleDriver	jdbc:oracle:thin:@<host>:1521:ora11g
Oracle	11.1.0.6.0 (11g)	JDK1.5: ojdbc5.jar; JDK1.6: ojdbc6.jar; classes12.jar	oracle.jdbc.OracleDriver	jdbc:oracle:thin:@//<host>:1521/ora11gsn (Oracle JDBC Thin using a ServiceName)
Oracle	11.1.0.6.0 (11g)	JDK1.5: ojdbc5.jar; JDK1.6: ojdbc6.jar; classes12.jar	oracle.jdbc.OracleDriver	jdbc:oracle:thin:@ora11gtn (Oracle JDBC Thin using a TNSName)
Oracle	10.1.0.2.0 (10g)	JDK1.2&1.3: classes12.zip; JDK1.4: ojdbc14.jar	oracle.jdbc.driver.OracleDriver	jdbc:oracle:thin:@<host>:1521:ora9i
Oracle	9i	JDK1.2&1.3: classes12.zip; JDK1.4: ojdbc14.jar	oracle.jdbc.driver.OracleDriver	jdbc:oracle:thin:@<host>:1521:ora9i
Oracle	8.1.7.0.0	classes12.zip	oracle.jdbc.driver.OracleDriver	jdbc:oracle:thin:@<host>:1521:userdb
DB2	9.7.200.358	db2jcc4.jar	com.ibm.db2.jcc.DB2Driver	jdbc:db2://<host>:50000/test
DB2	9.7.0.4	db2java.zip; db2jcc.jar; db2jcc_license_cu.jar; db2jcc4.jar; sqlj.zip; sqlj4.zip	com.ibm.db2.jcc.DB2Driver	jdbc:db2://<host>:50000/test
DB2	9.1.0.356	db2java.zip; db2jcc.jar; db2jcc_javax.jar; db2jcc_license_cu.jar; db2policy.jar	com.ibm.db2.jcc.DB2Driver	jdbc:db2://<host>:50000/test
DB2	8.2	Db2jcc.jar; db2jcc_license_cu.jar (Linux, Unix and Windows); db2jcc_license_cisuz.jar (Linux, Unixand Windows, z/OS, OS/390,z/OS, iSeries, and so on)	com.ibm.db2.jcc.DB2Driver	jdbc:db2://<host>:50000/test
DB2	8.2	Db2java.zip	com.ibm.db2.jdbc.app.DB2Driver	jdbc:db2:test
DB2	8.1.9.917	db2java.zip; db2jcc.jar; db2jcc_javax.jar; db2jcc_license_cisuz.jar; db2jcc_license_cu.jar	com.ibm.db2.jcc.DB2Driver	jdbc:db2://<host>:50000/test
DB2	8.1.7.380	db2java.zip; db2jcc.jar; db2jcc_javax.jar; db2jcc_license_cisuz.jar; db2jcc_license_cu.jar	com.ibm.db2.jcc.DB2Driver	jdbc:db2://<host>:50000/test
DB2	8.1	Db2java.zip; db2jcc.jar	com.ibm.db2.jdbc.net.DB2Driver	jdbc:db2://<host>/test
DB2	8.1	Db2java.zip	com.ibm.db2.jdbc.app.DB2Driver	jdbc:db2:test
RedBrick warehouse		redbrick.jar	redbrick.jdbc.RBWDriver	jdbc:rbw:protocol:<host>:5050/test/
Informix	11.70.TC7DE	ifxjdbc.jar	com.informix.jdbc.IfxDriver	jdbc:informix-sqli://<host>:9088/demo:INFORMIXSERVER=ol_informix1170
Informix	11.50.TC7DE	ifxjdbc-g.jar; jdbc.jar	com.informix.jdbc.IfxDriver	jdbc:informix-sqli://<host>:9090/informixdatatype:INFORMIXSERVER=ol_informix
Informix	9.40.TC1E1	ifxjdbc.jar	com.informix.jdbc.IfxDriver	jdbc:informix-sqli://<host>:1527/Demo:INFORMIXSERVER=ol_informix
Informix	9.30	ifxjdbc.jar	com.informix.jdbc.IfxDriver	jdbc:informix-sqli://<host>:1526/stores_demo:informixserver=DBSC
HSQL		hsqldb.jar	org.hsqldb.jdbcDriver	jdbc:hsqldb:D:\JReport\Demo\db\SampleDB
Sybase	12.5.2	jconn2.jar	com.sybase.jdbc2.jdbc.SybDriver	jdbc:sybase:Tds:<host>:5000/master
Sybase	12.5	jconn2.jar	com.sybase.jdbc2.jdbc.SybDriver	jdbc:sybase:Tds:<host>:5000/master
Sybase	12.5	jconn3d.jar	com.sybase.jdbc3.jdbc.SybDriver	jdbc:sybase:Tds:<host>:5000/master
Sybase	11.5	jconn2.jar	com.sybase.jdbc2.jdbc.SybDriver	jdbc:sybase:Tds:<host>:5000/master
Sybase IQ	15.4.0.3019	jconn3.jar; jconn4.jar	com.sybase.jdbc3.jdbc.SybDriver	jdbc:sybase:Tds:<host>:2638/iqdemo
PostGre SQL	8.3.0	postgresql-8.3-607.jdbc2.jar	org.postgresql.Driver	jdbc:postgresql://<host>:5432/postgres
PostGre SQL	8.2.13	postgresql-8.2-506.jdbc3.jar	org.postgresql.Driver	jdbc:postgresql://<host>:5432/postgres
PostGre SQL	8.0	postgresql-8.0-310.jdbc3.jar	org.postgresql.Driver	jdbc:postgresql://<host>:5432/test
Cache	Cache 4	CacheDB.jar	com.intersys.jdbc.CacheDriver	jdbc:Cache://<host>:1972/samples
Derby	10.8.1.2	derby.jar; derbyclient.jar	org.apache.derby.jdbc.ClientDriver	jdbc:derby://<host>:1528/test
Derby	10.5.3.0	derby.jar; derbyclient.jar	org.apache.derby.jdbc.ClientDriver	jdbc:derby://<host>:1527/test
Derby	10.5.1.1	derby.jar	org.apache.derby.jdbc.EmbeddedDriver	jdbc:derby:D:\derby\demo\databases\toursdb
MongoDB	2.2.2	already within JReport	toolkit.db.mongo.MongoDriver	host, port, databaseName:demo
HIVE	0.10.0	hadoop-common-2.0.0-cdh4.1.1.jar; hadoop-core-2.0.0-mr1-cdh4.1.1.jar; hive-exec-0.9.0-cdh4.1.1.jar; hive-jdbc-0.9.0-cdh4.1.1.jar; hive-metastore-0.9.0-cdh4.1.1.jar; hive-service-0.9.0-cdh4.1.1.jar; libfb303-0.7.0.jar; libthrift-0.7.0.jar; slf4j-api-1.6.1.jar; slf4j-simple-1.6.1.jar	org.apache.hadoop.hive.jdbc.HiveDriver	jdbc:hive://<host>:10000
PSQL	V11 SP3	jpscs.jar; pvjdbc2.dll; pvjdbc2.jar; pvjdbc2x.jar	com.pervasive.jdbc.v2.Driver	jdbc:pervasive://<host>:1583/test
Vertica		vertica-jdk5-6.1.2-0.jar	com.vertica.jdbc.Driver	jdbc:vertica://<host>:5433/vmartdb
Amazon RDS		mysql-connector-java-5.0.4-bin.jar	com.mysql.jdbc.Driver	jdbc:mysql://jrdbtest.c4fb8hiicidz.us-west-2.rds.amazonaws.com:3306/sampledb1110 (The URL is dynamically generated when you apply an instance.)
RedShift		postgresql-8.4-703.jdbc4.jar	org.postgresql.Driver	jdbc:postgresql://jinfonet-rsdw-demo.cfcn5ogc14yr.us-east-1.redshift.amazonaws.com:5439/sampledb (The URL is dynamically generated when you apply an instance.)
Hortonworks	0.14.0	hive-jdbc-0.14.0-standalone.jar; hadoop-common-2.6.0.jar	org.apache.hive.jdbc.HiveDriver	jdbc:hive2://<host>:10000/default;auth=noSasl

Notes:

- If you want to use the DB2 app connection, you need to install the client and configure the net address first.
- The database MySql with the example URL `jdbc:mysql://db06:3306/test?p?useSSL=true?clientCertificateKeyStoreUrl=D:\test\SSL_Client\ca-cert.pem?clientCertificateKeyStorePassword=1234` is connected by SSL. For how to install SSL, refer to http://www.openssl.org. For how to configure MySql for SSL, refer to http://dev.mysql.com/doc/refman/5.1/en/ssl-connections.html. For more information about JDBC driver, refer to http://dev.mysql.com/doc/connector-j/en/connector-j-reference-configuration-properties.html.
- For Oracle JDBC Thin using a TNSName, you need to configure the tnsname first, then add the parameter `-Doracle.net.tns_admin=<designer_install_root>\lib` into the file JReport.bat which is located in `<designer_install_root>\bin`, assuming you have copied the file tnsnames.ora from your Oracle server to `<designer_install_root>\lib`.

Installing using the Installation Wizard

Installing JReport Server with the Installation Wizard is intuitive. You only need to follow the screens and enter the required information. The Installation Wizard provides two installation types:

- **Typical Installation for Standalone Server**
  Installs JReport Server with the default configuration settings.
- **Custom Installation for Standalone Server**
  Installs JReport Server in a standalone environment. If you choose this installation type, you can configure the server system environment in the Installation Wizard.

This document shows you how to install the server to different systems with the Installation Wizard and configure the server according to your requirements.

Installing on Windows

To install JReport Server on a Windows platform, take the following steps:

1. Download the JReport Server installation file for Windows from the Jinfonet download center: http://www.jinfonet.com/downloadjreport/.
2. Run the installation file and follow the prompts to install.

During installation, pay attention to the following:

- The installer requires that you choose a Java JDK to complete the installation. You can download the appropriate JDK from http://java.sun.com or your computer vendor's website.
- The Installation Wizard will first find a JVM to get started. If no JVM is found, the JReport installer will fail to launch. To solve this issue, you can try either way:
  - Set JAVA_HOME variable and append `%JAVA_HOME%\bin` to Path variable in system environment.
  - Install JReport Server from a DOS command by specifying the `LAX_VM` option for the Installation Wizard as follows:

    `jrserver-xxx-windows.exe LAX_VM "C:\jdk1.6.0_17\bin\java.exe"` (change *jrserver-xxx-windows.exe* to the real file name of the installation file)

    The JDK path should use absolute path and be quoted by "".
- The installer provides a chance for you to add additional class paths. You can also choose to add them manually into the setenv.bat in `<install_root>\bin` after installation.

Installing on Unix

JReport Server supports Solaris, Linux, HP-Unix, and AIX. In the following process, an X server is running and Java 1.6 or above is available, otherwise ask your administrator for help. Installing and running JReport Server requires that an X server has been configured.

1. Download the JReport Server installation file for Unix from the Jinfonet download center: http://www.jinfonet.com/downloadjreport/.

   If you need to transfer the installation file from your download machine to your Unix box, you should transfer it using FTP in binary mode.
2. Click the installation file to launch the Installation Wizard. Alternatively, you can open a console window, and change the directory to the location of the file. Following are examples of the commands that can be used:

   `$ cd /opt/JReport/Server` (or your preferred install location)

   To make the installation file executable, type the command:

   `$ chmod +x jrserver-xxx-linux.bin` (change *jrserver-xxx-linux.bin* to the real file name of the installation file)

   To run the installation file:

   `$ ./jrserver-xxx-linux.bin` (change *jrserver-xxx-linux.bin* to the real file name of the installation file)

   The Installation Wizard will first locate a JVM to get started. If no JVM is found, the installer will fail to launch. To solve this issue, you can try either way:
   - Set JAVA_HOME variable and append `$JAVA_HOME/bin` to PATH variable in system environment.
   - Specify a JVM for Installation Wizard with the option `LAX_VM` as follows:

     `$ ./jrserver-xxx-linux.bin LAX_VM "/opt/jdk1.6.0_17/bin/java"` (change *jrserver-xxx-linux.bin* to the real file name of the installation file)

     The JDK path should use absolute path and be quoted by "".
3. Once the Installation Wizard has successfully loaded, you can follow the standard prompts to install JReport Server.

Installing on z/Linux

JReport Server supports Linux on IBM system z. In the following process, an X server is running and a JDK specially used for IBM is available, otherwise ask your administrator for help. Installing and running JReport Server requires that an X server has been configured.

1. Download the JReport Server installation file for z/Linux from the Jinfonet download center: http://www.jinfonet.com/downloadjreport/.

   If you need to transfer the installation file from your download machine to your z/Linux box, you should transfer it using FTP in binary mode.
2. Click the installation file to launch the Installation Wizard. Alternatively, you can open a console window, and change the directory to the location of the file. Following are examples of the commands that can be used:

   `$ cd /opt/JReport/Server` (or your preferred install location)

   To make the installation file executable, type the command:

   `$ chmod +x jrserver-xxx-linux.bin` (change *jrserver-xxx-linux.bin* to the real file name of the installation file)

   To run the installation file:

   `$ ./jrserver-xxx-linux.bin` (change *jrserver-xxx-linux.bin* to the real file name of the installation file)

   The Installation Wizard will first locate a JVM to get started. If no JVM is found, the installer will fail to launch. To solve this issue, you can try either way:
   - Set JAVA_HOME variable and append `$JAVA_HOME/bin` to PATH variable in system environment.
   - Specify a JVM for Installation Wizard with the option `LAX_VM` as follows:

     `$ ./jrserver-xxx-linux.bin LAX_VM "/opt/ibm-java2-sdk-6.0/bin/java"` (change *jrserver-xxx-linux.bin* to the real file name of the installation file)

     The JDK path should use absolute path and be quoted by "".
3. Once the Installation Wizard has successfully loaded, you can follow the standard prompts to install JReport Server.

Installing to an existing JReport Server directory

If you select to install JReport Server in a folder that already contains an existing JReport Server, you can choose to continue the installation in either the upgrading way or overwriting way. If you choose the upgrading way, the installer will replace the packages and create new batch/script files. Meanwhile, a copy of the old batch/script files will be kept for your reference. You should use the batch/script files that come with the installer in order to make sure that all new packages are added to the class path and manually merge any changes you made into the new version. However, your configuration files and server runtime data of the existing copy will be kept, and you can choose to recover the previous version during the uninstallation of the newly installed JReport Server.

Configuring system database

Configuring the system database is a step on the server installation wizard available to a production key or a temporary key.

By default the trial database Derby is used. The port is configurable in case it is being used.

However in a production environment, you'd better configure your own production DBMS instead of using the default Derby which is provided for testing and evaluation purposes only and which should not be used in a production system. To do this, select **Production Database** and then set the correct database connection information. If the connection failed after clicking Next, a message will pop up, you can choose to reset the connection. Or if you choose to continue with the failure, after the installation, you need to configure the database still. For details, see Configuring the server database.

These databases have been tested workable as the production database: HSQLDB, MySQL, Microsoft SQL Server, IBM DB2, Oracle, Sybase, and Informix.

Notes:

- ODBC is not supported as the server database.
- When setting SQL 2000 as the server database, the driver should use jtds.jar, otherwise the scheduling feature cannot work.

Configuring system environment

When installing JReport Server using the Installation Wizard, if you choose Custom Installation for Standalone Server, you can configure the server system environment according to your requirements during the installation. You can configure the following items:

- Service
- Cluster
- E-mail
- **Cache**
  - **Cache Loaded Catalogs**
    Specifies whether to keep a catalog in memory, or to remove it from memory after a report is completed.

    Normally, after a report has been generated, the catalog that is used to generate the report will be removed from memory. However, if you specify this option, the catalog will be cached rather than removed.

    Cache Loaded Page Reports
    Specifies whether to keep the page reports in memory or remove them from memory after they have been generated.
- **Performance**
  Pre-loading the Java classes and fonts which are used by catalogs, reports and JReport Engine at startup time will improve performance when these classes are needed at runtime.
  - **Preload Catalog Referred Classes**
    Pre-loads the Java classes that are used for a catalog.
  - **Preload Page Report Referred Classes**
    Pre-loads the Java classes that are used for a report.
  - **Preload Engine Referred Classes**
    Pre-loads the Java classes that are used by JReport Engine to generate reports.
  - **Preload Fonts**
    Specifies to load the fonts when JReport Server is started rather than when they are first used.
  - **Maximum Number of Concurrent Reports in the Queue**
    Specifies the maximum number of concurrent reports in the queue, which must be less than or equal to the number that the license permits. For details, see Appendix 1: Properties in the server.properties file.
- Advanced

Installing silently

JReport provides two files for installing JReport Server silently without user participation in the installation process, which are ServerInstall_typical.properties for the Typical Installation for Standalone Server type and ServerInstall_custom.properties for the Custom Installation for Standalone Server type.

Follow the steps below to install JReport Server silently:

1. Copy the appropriate file from `<install_root>\help\samples\SilentInstall` to the directory where the JReport Server installation file is saved and edit it to suit your requirements.

2. Some built-in demo reports (*.cls) with Derby as the data source (install_root\db\SampleDB.script) have been provided. At the end of the installation, the installer will configure the reports and catalog to the correct data path. This calls some AWT classes that require GUI support.

   So, if you have an X server installed, you should set the Display variable so that this step can be performed successfully.

   ```
   $ DISPLAY=hostname(or IP address):0.0
   $ export DISPLAY
   ```

   > **Note:** If you do not have X server or a pure text environment, this step can be ignored. However, you may find that the demo reports will not be able to run after you start the JReport Server due to having the wrong default data source path. In this case, you can use JReport Designer to publish some working reports for testing purposes.

3. Run the following command, and JReport Server will be installed in the designated path:

   `$ ./jrserver-xxx-linux.bin -i silent -f ServerInstall_typical.properties` (change *jrserver-xxx-linux.bin* to the real file name of the installation file)

Notes:

- When installing JReport Server silently, make sure you do not use upgrade installation, it is not supported for now.

- When you install the Update or Service Pack silently, edit the file update.properties in `<install_root>\help\samples\SilentInstall` to your own requirements. This file is used to create an option file (that is, response file) for the Installation Wizard. It predefines all the information that is required for the installation.

  You can also create a property file and save it as follows:

  ```
  USER_INSTALL_DIR=/usr/local/JReport/Server
  USER_KEY=UID
  USER_PASSWORD=Password
  ```

  Modify the above lines according to your own environment and configurations.

Installing using the console interface

JReport enables you to perform an interactive installation from a command prompt on platforms that do not have GUI. In this way, you are able to see the installation status and follow the installation process.

Take the following steps to perform console installation:

1. Download the appropriate installation file according to your system reuqirement from the Jinfonet download center: http://www.jinfonet.com/downloadjreport/.

2. Run the following command:

   For Unix and z/Linux:

   `$ chmod +x jrserver-xxx-linux.bin`

   `$ ./jrserver-xxx-linux.bin -i console` (change *jrserver-xxx-linux.bin* to the real file name of the installation file)

   For Windows:

   `jrserver-xxx-windows.exe -i console` (change *jrserver-xxx-windows.exe* to the real file name of the installation file)

3. Make decisions following the installation process.

Installing on Unix manually

In some rare cases, JReport Server may fail to install on Unix directly. In this case, follow the steps below to install it manually:

1. Install JReport Server on Windows following the steps in Installing on Windows, but don't start it.
2. Prepare a directory on Unix where you will copy the installation, for example `/opt/JReport/Server`.
3. Modify javahome and reporthome in the following files in `<install_root>\bin` to the Unix directories where the Java JDK is located and the directory you are going to copy the release to, using absolute path. Be sure to modify them carefully. Any mistake will cause problems starting JReport.

   report.ini
   servlet.properties
   setenv.sh
4. Modify javahome in the file env.sh in `<install_root>\derby\bin` to the Unix directory where the Java JDK is located, using absolute path.
5. Delete the file server.properties from `<install_root>\bin` if it exists and remember to reset the required configuration settings on the JReport Administration page (8889 as the default port) after launching JReport Server on Unix. The server.properties file is created if you use the custom format to install.
6. Make a zip or jar archive of the above folders, and then copy it to your Unix system (use binary format if using FTP).
7. Extract the folder in the destination directory in accordance with the path defined in the property files.
8. Use the `dos2unix` command to convert all the .sh files under `<install_root>/bin` to the format that can be recognized by Unix. You can execute the command like this:

   ```
   $ dos2unix *.sh
   ```
9. Use the `chmod` command to set the converted files under `<install_root>/bin` to have read, write and execute permission. You can execute the command like this:

   ```
   $ chmod 777 *.sh
   ```
10. Start a shell (Console) and login as root or become the root user by running the `su` command. Make JRServer.sh executable and then start JReport Server by running `./JRServer.sh`.

Note: If you fail to intall JReport Server on your z/Linux system directly, you can also follow the above steps to install the server manually.

Installing by building a war file on Windows to deploy to Linux/Unix

There are user cases when they need to install JReport Server into application servers in the cloud and may not even own a machine of the type they are installing to. For example you use Tomcat on Linux in a cloud environment but only have Windows machines available locally. You can then install JReport server on Windows, make a war file and then deploy it to the Tomcat on Linux.

1. Install JReport Server on the local Windows machine.
2. Create a WAR via the following command line while specifying a report home. The report home uses the path on Linux on the target machine.

   ```
   makewar.bat -Dreporthome=/opt/reporthome
   ```

3. Add any JDBC drivers that are required into the `jreport.war/WEB-INF/lib` directory.
4. Visit Tomcat from the local machine by the URL `http://<host>:8080/`, deploy the WAR. 8080 is the default port of Tomcat and suppose it works.
5. Log onto JReport Server Administration console by the URL `http://<host>:8080/jreport/admin`, go to the Data tab on the system toolbar, configure the server database for System DB/realm DB/ Profiling DB. It shouldn't be Derby.
6. Restart JReport Server and access `http://<host>:8080/jreport` to see if you can run reports.

Uninstalling JReport Server

Use either of the following methods to remove JReport Server:

- From the Control Panel, open **Add or Remove Programs**, and then select **JReport Server 13.5 Update 2** to uninstall it.
- Run uninstaller.exe (uninstaller on Unix) in `<install_root>\_uninst`.

Note: The uninstaller will remove all the files generated by the installer, while the files that are created later by the program will be retained. They should be removed manually.

Solving installation problems

This section is trying to help you solve the problems you encounter during the installation.

Where to find log information

If error occurs during the installation, you can check the log information recorded to find out what the problem is. Where logs are generated depends on when the installation process get stuck:

- If the installation is cancelled before you click the Install button on the installation wizard, logs are created on the desktop for Windows and in the userhome directory for Unix/Linux.
- If the installation is cancelled after you click the Install button on the installation wizard, logs are created in the logs folder in the installation root directory.

Besides, on a Windows platform, you can choose to specify the log destination that should use absolute path and log file name when launching the installation wizard by running the following command:

```
jrserver-xxx-windows.exe -D$INSTALL_LOG_NAME$="Install.log" -D$INSTALL_LOG_DESTINATION
$="D:\temp"
```

or

```
$ ./jrserver-xxx-linux.bin -D$INSTALL_LOG_NAME$="Install.log" -D$INSTALL_LOG_DESTINATION
$="/opt/temp"
```

Change *jrserver-xxx-windows.exe* or *jrserver-xxx-linux.bin* to the real file name of the installation file.

Feel free to send your questions to support@jinfonet.com.

An issue on Windows Vista

Problem

When running JReport Server's installer on Windows Vista, the installer cannot find the installed JDKs.

Reason

By default Vista's security settings are stricter than Windows 2000, XP, and 2003. The children processes do not inherit the execution right from their parent process.

Solution

Make the compatibility property of the installer file (.exe) available:

1. Right-click the installer file, and select **Properties** from the shortcut menu.
2. In the Compatibility mode panel of the Compatibility tab, check the **Run this program in compatibility mode for** option, and then select **Windows 2000** from the drop-down list.
3. Click **OK**.
4. Run the installer.

JReport Server reporthome directory overview

This section provides a general view of the directories in the JReport Server installation root, including what they contain, what they are used for, and how to set their location if possible.

The following is a list of the server reporthome directories:

Directory	Contents	Directory location Configurability
_uninst	Files used for uninstalling the JReport Server.	Fixed.
bin	Command, configuration, and properties files.	Fixed.
db	Demo reports' database.	Fixed.
Demo	Demo reports used in the tutorial lessons.	Fixed.
derby	The Derby program and database resources.	Fixed.
dynamicclasses	UDS jar/zip files.	The directory location can be specified by the server.dynamic.class.dir property in the server.properties file in `<install_root>\bin`.
font	TTF font.	The directory location can be specified by -Djreport.server.font.path or by the server.font.path property in the server. properties file in `<install_root>\bin`.
gisinfo	Report related Geographic Information files.	Fixed.
help	Help documents introducing the function, features and usage of JReport Server and JReport Designer together with Tutorial manual.	Fixed.
history	Version files.	The directory location can be specified by the servlet.jrserver.initArgs property in the servlet.properties file in `<install_root>\bin`.
images	Public images for Page Report Studio.	Fixed.
jreports	Demo reports. When scheduling a task to disk, the directory refers to the destination root of the server resource tree.	Fixed.
lib	Library files required by JReport runtime.	Fixed.
logs	Log files.	The directory is the default location for log files.
ntservice	Files for C program and for writing a Windows NT-service to run JReport Server.	Fixed.
portlet	Resources for building portlet wars.	Fixed.
prestart	Reads customized configuration for launching JReport Console and Administration pages from the Start menu.	Fixed.
profiling	Profiling related files.	Fixed.
properties	Default location for JReport Server realm database.	The directory location can be specified using the URL option on the JReport Administration page > Data > Realm DB > Configuration tab.
public_html	Standalone web app folder.	Fixed.
realm	Realm files.	Fixed.
resources	Language packages for specifying JReport Server UI language.	Fixed.
scratchdir	Output files of compiled JSPs.	The directory location can be specified by the servlet.jspservlet.initArgs property in the servlet.properties file in `<install_root>\bin`.
script_files	Script files for creating and deleting system database tables.	Fixed.
style	CSS style files and style group files.	The directory location can be specified by stylePath in the report.ini file in `<install_root>\bin`.
temp	Engine temp files and Server temp result files.	For Engine temp files, the directory location can be specified by tempPath in the report.ini file in `<install_root>\bin`. Fixed for Server temp result files.
templates	Templates for web reports.	Fixed.
txtdriver	Demo reports' flat data files.	Optional.

Upgrading JReport Server

Upgrading a standalone JReport Server and an integrated JReport Server involves different processes while sharing some common steps. Both updated JReport Servers require converting the report resources in the old version in order to comply with the new server version.

Before upgrading, the old version JReport Server should be shut down.

Upgrading in a standalone environment

JReport Server 13.5 provides two migration tools in `<install_root>\bin` folder, which can help you to convert all the resources on the previous version of JReport Server before version 6.0. Resources the migration tools cover include the security information (realm, user, group, protection, and ACL), report resources (catalog and reports), scheduled tables, completed tables, version and version tables, and other relevant information (such as fonts, NLS, and style groups).

Note: When installing the JReport Server 13.5 into the same directory as the old version, the report level resources in the old version will be maintained and not be replaced by the report resources of V13.5. For example, if you installed JReport Server 13.5 to `C:\JReport\Server13GA`, the backed up files of V13 GA will be maintained in `C:\JReport\ServerV13GA\bak\previous_version`, the removed patches and jar files will be maintained in `C:\JReport\ServerV13GA\bak\xxx\lib`, and the whole folder bin will be maintained in `C:\JReport\ServerV13GA\bak\xxx\bin`.

Upgrading a version later than V13 GA (included) to V13.5

To upgrade a version later than V13 GA (included) to V13.5, take the steps below. Here is a case showing you how to upgrade V13 GA to V13.5, provided that the old JReport Server is located in `C:\JReport\Server13GA`.

1. Install JReport Server 13.5 to `C:\JReport\Server13GA`. In the Choose Installation Set screen, choose **Typical Installation** for Standalone Server and click **Next**.
2. In the prompted Installation Warning message window, choose **Upgrading**.
3. The installation turns to the List screen which shows the details of the upgrading, including the removed patches, the modified config files and the backup files.
4. Click **Install** in the Installation Summary screen. After the installation is complete, start JReport Server to check the upgraded version.

Upgrading a version later than V6.0 (included) to V13.5

To upgrade a version later than V6.0 (included) to V13.5, take the steps below. Here is a case showing you how to upgrade V11.1 to V13.5, provided that the old JReport Server is located in `C:\JReport\Server11.1`.

1. Check the value of properties in dbconfig.xml which locates in `C:\JReport\Server11.1\bin` or back up the file firstly.
2. Install JReport Server 13.5 to `C:\JReport\Server11.1`. In the Choose Installation Set screen, choose **Typical Installation for Standalone Server** and click **Next**.
3. In the prompted Installation Warning message window, choose **Yes**.
4. In the System Database screen, input the property values according to the backed up dbconfig.xml file. Or check the **Trial Database** radio button, and replace dbconfig.xml file with the one you backed up.
5. In the List screen you can see the details of the upgrading, including the removed patches, the modified config files and the backup files.
6. Click **Install** in the Installation Summary screen. After the installation is complete, start JReport Server to check the upgraded version.

Upgrading a version between V5.2 Build 590 (included) and V6 (excluded) to V13.5

You can make the upgrade using the migration tool MigrationV52.bat (MigrationV52.sh for Unix) that is available in the `<install_root>\bin` folder. This tool is used to convert all the resources of the versions between V5.2 Build 590 (included) and V6 (excluded) to the resources of JReport Server 13.5. If you install the new version to the same folder as the old one, the parameter can be omitted.

- **Usage**
  MigrationV52 [orgReportHome]
- **Options**
  - orgReporthome
    The reporthome of the original JReport Server. If this parameter is not provided, the reporthome of JReport Server 13.5 will be used as its value.

Case 1: Installing JReport Server 13.5 to a new folder (recommended)

1. Provided that the old JReport Server is located in `C:\JREntServer595`. Install JReport Server 13.5 to a new folder `C:\JReport\ServerV13.5`. DO NOT start the newly installed JReport Server.
2. In the DOS window, switch to `<install_root>\bin`, and run `MigrationV52 C:\JREntServer595`.

Case 2: Installing JReport Server 13.5 to the folder where the old version resides

1. Provided that the old JReport Server is located in `C:\JREntServer595`. Install JReport Server 13.5 to the same location. DO NOT start the newly installed JReport Server.
2. In the DOS window, switch to `<install_root>\bin`, and run `MigrationV52.bat`.

Upgrading a version earlier than V5.2 Build 590 (excluded) to V13.5

You can do the upgrade using the migration tool MigrationBV52.bat (MigrationBV52.sh for Unix) that is available in the `<install_root>\bin` folder. This tool is used to convert all the resources of versions lower than V5.2 Build 590 to the resources of JReport Server 13.5. If you install the new version to the same folder as the old one, the parameter can be omitted.

- **Usage**
  MigrationBV52 [orgReportHome]
- **Options**
  - orgReporthome
    The reporthome of the original JReport Server. If the parameter is not provided, the reporthome of JReport Server 13.5 will be used as its value.

Case 1: Installing JReport Server 13.5 to a new folder (recommended)

1. Provided that the old JReport Server is located in `C:\JREntServer580`. Install JReport Server 13.5 to a new folder `C:\JReport\ServerV13.5`. DO NOT start the newly installed JReport Server.
2. In the DOS window, switch to `<install_root>\bin`, and run `MigrationBV52 C:\JREntServer580`.

Case 2: Installing JReport Server 13.5 to the folder where the old version resides

1. Provided that the old JReport Server is located in `C:\JREntServer580`. Install JReport Server 13.5 to the same location. DO NOT start the newly installed JReport Server.
2. In the DOS window, switch to `<install_root>\bin`, and run `MigrationBV52.bat`.

Upgrading in an integration environment

1. Suppose that you already have a standalone JReport Server 13.5, whether it is installed directly or upgraded from a previous version. In this step you need not convert the old version reports.
2. Use the JReport Server 13.5 to create a self-contained WAR/EAR file. For information on how to create the WAR/EAR, see Building a WAR/EAR file to include a self-contained JReport Server. Remember that when building the jreport.war file in a standalone JReport Server 13.5, the ReportHome of jreport.war must be the same as that of the jreport.war in the previous version.
3. Use the Java application server that integrates the previous version of JReport Server to deploy the new self-contained WAR/EAR and update the old WAR/EAR with the new one.

   By now, if the previous version of JReport Server is later than V6.0 (included), the integrated JReport Server has already been upgraded to V13.5. But if the previous version is earlier than V6.0, go to step 4 and you need do further upgrade.
4. For a previous version between V5.2 Build 590 (included) and V6 (excluded): in the DOS window, switch to `<reporthome>\bin`, and run `MigrationV52.bat`.

   For a previous version earlier than V5.2 Build 590 (excluded): in the DOS window, switch to `<reporthome>\bin`, and run `MigrationBV52.bat`.
5. Clear the temporary files or cached files on the application server. Take Tomcat as an example, remove the *temp* and *work* directories in `D:\Tomcat`.

Launching JReport Server

After you have installed JReport Server, you must set up the reporting environment and start it before you can access it via a web browser. This chapter provides you with detailed information of how to run JReport Server in different running modes.

The following topics are discussed in this chapter:

- Setting up the reporting environment
- Running as a standalone server
- Running as an OS service
- Running within an application server

Setting up the reporting environment

Now that you have installed JReport Server, you can start it. However, it is better to first check the reporting environment to see whether you have published the reports including the catalog files, added the necessary class paths, and set up the data sources.

Report publishment & creation

The separation of the report file and the data is powerful. It facilitates easy re-use of the report's layout. Report files built with JReport Designer can then be published to JReport Server for running in the thin-client/server mode. Today, computing is a service-based model in which web based applications are rapidly replacing monolithic fat-client hosted and maintained software applications. JReport Server is a high performance server for running reports on demand or unattended on a scheduled basis. In this environment, thousands of clients may view, print and generate new reports.

JReport Server has its own resource tree with each node mapping to a folder or file in your physical drive. So when publishing report files to the server, it is recommended that you check:

- Whether the report files reside somewhere in the server machine.
- Whether the files are mapped in the resource tree.
- That not only the report files, but also the catalog files (including the .fml file) exist and are in the same directory. In addition, one directory can only contain one catalog. However, it can contain many report files which use that catalog.

Data sources

You may have your own catalogs and reports that you developed, and want them to be run and be distributed by JReport Server. In order to do this, the data sources used by your catalogs must be correctly set to the runtime environment of JReport Server.

User data source

Append the class path of your user class files to the first item of the classpath set in your batch file or command line that starts JReport Server. If you are using JReport Server embedded in an application server, add your user class files to the WAR file used to deploy JReport Server.

Additional class paths

If your reports reference any external classes, you will need to add them to the classpath option in the batch file and command line that starts JReport Server. For example, if your reports contain user defined objects (UDO) or user defined formula functions, do not forget to add them to the class path or to the WAR file.

Running as a standalone server

JReport Server can be started from the web application server contained in the JReport Server package. This section explains how to start JReport Server as a standalone report server, how to send commands to the server and how to use the batch files in the `<install_root>\bin` directory.

- Starting using launch files
- Starting using Java
- Sending commands to JReport Server from Java

Starting using Java

The class of the standalone server is jet.server.JREntServer. You can start JReport Server with the following command instead of using the generated batch files:

```
JAVA -classpath <classpath> -Djava.compiler=NONE -Dreporthome=<install_root> jet.server.
JREntServer [options]
```

- **-classpath**
  The classpath must include the following packages originally in your `<install_root>\lib`: JRESServlets.jar; JREntServer.jar; JREngine.jar; servlet.jar; log4j-1.2.8.jar;
- **-Djava.compiler=NONE**
  This is without JIT. This is not a required option. However, if you encounter problems running the server and you think that they relate to the Java VM, you can try turning off the JIT compiler and then running again.
- **-Djreport.url.encoding**
  Specifies the encoding to encode/decode escape characters in URL strings. If not specified, the system default encoding will be used. For example: java ... -Djreport.url.encoding=8859-1...
- **-Dreporthome**
  This is where JReport Server is installed. It is the Destination Location you specified when you installed it. This option is required. When you set the reporthome, upon launching, JReport will try to find the jslc.dat and report.ini files in `<install_root>\bin` and check whether they are valid. Jslc.dat is the License control file. Open report.ini, and you will find the configuration information, including the temp, template and the help path. JReport will use the temp path to export the temporary files so you should make sure that the temp folder specified in report.ini exists and has space available.
- **-Dfile.encoding**
  Specifies the encoding to encode/decode escape characters in the server data. If not specified, the system default encoding will be used. For example: java ... -Dfile.encoding=8859-1...
- **-Dresolution**
  Sets the system resolution in DPI. If not specified, the system default resolution will be used, which is the resolution of your monitor, for example, -Dresolution=96.
- [**options**]

Option	Description
-?	Print brief help message.
-p port	The port that this server listens on, default is 8888.
-ap adminport	The port number that the remote administration uses, default is 8889.
-l backlog	Maximum length of queue for incoming connection indications.
-m max	Maximum number of connection handlers.
-t timeout	Connection timeout in milliseconds.
-s filename	Servlet property file name.
-realm realmname	Active realm when the server starts up. The specified realm should exist, otherwise the server will use an existing realm as the active realm. The server will then record a warning message in the log file, and set the selected active realm by the server.realm.active property in the server.properties file.
-web directory	Web application server root directory, default is `<intall_root>\public_html`.
-local	Administration on local host only.
-vDebug	Enables JReport Engine to output messages to a file and sets engine log file's trace level to INFO and error level to WARN.
-vError	Enables JReport Engine to output messages to a file and sets engine log file's trace level to OFF and error level to ERROR.
-env	Print environment settings when the server starts up.
-silent	Outputs nothing, not even the server start information.
-log[:file Name] (deprecated)	Outputs JReport Engine messages to the log file as specified and uses the -vDebug level.
-logall	Sets all loggers' trace level to INFO and error level to WARN.
-jrs.admin.server host:port	The admin server host and RMI port.
-cleanup	Checks integrality of the server data and cleans up the invalid data.

Notes:

- For detailed information on how to configure the logging and debugging information, read the LogConfig.properties file in `<install_root>\bin`.
- Some of the common options will be used in later chapters. In addition, JReport has automatically generated some batch files for you so that you do not have to write a complicated command line. You can find these in the `<install_root>\bin` directory.

Sending commands to JReport Server from Java

After JReport Server has been started as a standalone server, you can send commands to the server to either shut it down or pop up the user interactive interface for administration. All of these can be done through the class jet.server.CommandSender. The full command is as follows:

```
JAVA -classpath <classpath> -Djava.compiler=NONE [-Dreporthome=<install_root>] jet.
server.CommandSender [-s:server -p:port] -w:password [-?]|admin|shutdown|gc|(local:on|
off)
```

- **-?**
  Prints brief help message.
- **-Djava.compiler=NONE**
  This is without JIT. This is not a required option. However, if you encounter problems running the server and you think that they relate to the Java VM, you can try turning off the JIT compiler and running again.
- **-Dreporthome**
  This is where JReport Server is installed. It is the destination location you specified when you installed it. It is required only if you do not execute the command from the local host on which JReport Server is running.
- **-Dpoperror=true**
  This property is used to control whether to pop up a message to show error information. The default value is false which indicates that the error message will not be displayed.
- **-Classpath**
  The classpath must include the following packages originally in your `<install_root>\lib`: JRESServlets.jar; JREntServer.jar.
- **-s:server**
  Host name on which JReport Server is running.
- **-p:port**
  The port JReport Server used for administration. The default value is 8889.
- **-w:password**
  Password of the admin user. Example: -w:admin.
- **admin**
  A command sent to the server asking to pop up the user interactive interface for administering JReport Server.
- **shutdown**
  A command sent to the server asking it to shut down.
- **local:on**
  A command sent to the server asking to only allow the administration commands sent by the local machine.
- **local:off**
  A command sent to the server asking to accept administration commands from anywhere.
- **gc**
  A command sent to the server asking the JVM to schedule the Java Garbage Collector.

Note: Some of the common options will be used in the later chapters. In addition, JReport has automatically generated the batch file CmdSender.bat for you so that you do not have to write a complicated command line.

Running as an OS service

JReport Server can be configured as an OS service. This section shows you how to run JReport Server as a service of Windows, Unix and Linux.

Running as a Windows service

The tool JRservice.exe saved in `<install_root>\bin` is provided to assist you in installing JReport Server as a Windows service.

You can get the following information if you run JRservice.exe without any options:

```
Usage:

JRService -install [-interactive] to install the service
JRService -remove                 to remove the service

    -interactive to enable JReport Server service
                 to interact with the desktop

StartServiceCtrlDispatcher being called.
This may take several seconds. Please wait.
```

- **JRService -install**
  Running JRservice.exe with the `-install` option will install JReport Server as a Windows service. If you open the Services item in the Control Panel, you will find a service named JReport Server in the list.
- **JRService -install -interactive**
  If you use the `-interactive` option together with `-install`, the service installed will be run in interactive mode. That is, when you start the service, a Command Prompt window will pop up. However, if you don't specify this option when you install the service, the Command Prompt window will not be displayed when you start the service. After you have installed JReport Server as a service, you will then be prompted to restart your computer for the service installation to take effect.
- **JRService -remove**
  Running JRservice.exe with the `-remove` option removes the Windows service of JReport Server from Windows. However, before you run this, you should stop the service.

Configuring the service

The parameters for the Windows service of JReport Server are read from the file NTService.ini in `<install_root>\bin`. In this file, you can find three parameters specified, as shown in the following example:

```
JavaVM="C:\jdk1.6.0_17\jre\bin\java.exe"
StartArg= "-Dinstall.root=C:\JReport\Server"
-classpath "C:\JReport\Server\lib\JRentServer.jar;
C:\JReport\Server\lib\JREngine.jar;C:\JReport\Server\lib\servlet.jar;C:\TEMP"...
-Djava.compiler=NONE -Dreporthome=C:\JReport\Server jet.server.JRentServer
ShutdownArg= "-Dinstall.root=C:\JReport\Server"
-classpath "C:\JReport\Server\lib\JRentServer.jar;
C:\JReport\Server\lib\JREngine.jar;C:\JReport\Server\lib\servlet.jar;C:\TEMP" ...
-Djava.compiler=NONE
-Dreporthome=C:\JReport\Server jet.server.CommandSender localshutdown
```

- **JavaVM**
  The path of the Java VM.
- **StartArg**
  The Java command line for launching JReport Server as an independent web application server. This will be called when the service is started.
- **ShutdownArg**
  The Java command line for shutting down JReport Server. This is called when the service is stopped.

Starting the service

There are two ways to start the JReport Server service.

- After the server has been installed as a service, it is by default configured to start automatically each time. In other words, without otherwise modifying the Control Panel, the service will start automatically each time Windows is started.
- You can directly start the service through the Services item in the Control Panel. Open the Services list, find JReport Server on the list, select it and click the **Start** button.

You can change the options in the file NTService.ini in `<install_root>\bin` before you start the service (for more information on the options available, see Starting JReport Server using Java). In the example above, there are no options specified in StartArg. If you would like to set all error log levels to WARN and trace log levels to INFO, you will need to append -logall at the end, as in the following example:

```
...
StartArg= "-Dinstall.root=C:\JReport\Server"
-classpath "C:\JReport\Server\lib\JRentServer.jar;
C:\JReport\Server\lib\JREngine.jar;C:\JReport\Server\lib\servlet.jar;
C:\JReport\Server\lib\log4j-1.2.8.jar;C:\TEMP" -Djava.compiler=NONE
-Dreporthome=C:\JReport\Server jet.server.JRentServer -logall
...
```

Stopping the service

There are three ways for you to stop the JReport Server service:

- Open the Control Panel, go to **Administrative Tools**, double-click the **Services** item, select **JReport Server**, and then click the **Stop** button if it is not disabled.
- Run the batch file CmdSender.bat in `<install_root>\bin` with the *localshutdown* argument, for example: `<install_root>\bin\CmdSender.bat localshutdown`.
- Use the Shut Down the Server button on the JReport Administration page.

Notes:

- All ODBC data sources used by the JReport Server Service belong to the System DSN. System data sources can be used by all users on a computer, and are visible to all users on the computer and system-wide services, such as Microsoft Windows services. User data sources can only be used by the current user and are visible only to that user. To establish JReport Server as a service, you should choose System Data Source. That is, define the data source in the System DSN. To do this, open Data Source (ODBC) in Control Panel, and add the data source used by JReport Server to the System DSN panel. Also, remove any old ones from the User DSN.
- When using NT service to start JReport Server, the mapped disk in path cannot be accessed due to JVM limitation. You should use UNC path (for example, `\\127.0.0.1\public_write`) instead of the mapped disk it is mapped to (for example, Z).
- When using the Local System account to run JReport Server as a Windows service, under some circumstances writing to a UNC path (for example, `\\127.0.0.1\public_write`) is not allowed. To solve this problem, you need to start the service by using a normal Windows user.

Running as a service on Unix

Assume that JReport Server has been installed to `/user/report/jns`,

1. Write a script `/etc/init.d/jrserver` as follows, and make it executable.
   ```
   #!/bin/sh
   mode=$1
   if [ ! -d /user/report/jns ]
   then # JReport not installed
   exit 1
   fi
   case "$mode" in
   'start')
   if [ -d /user/report/jns ]
   then
   echo "Starting JReport Server"
   /user/report/jns/bin/NJRServer.sh &
   fi
   ;;
   'stop')
   if [ -d /user/report/jns ]
   then
   echo "Stopping JReport Server"
   /user/report/jns/bin/CmdSender.sh localshutdown &
   fi
   ;;
   *)
   echo " Usage: "
   echo " $0 start (start JReport Server)"
   echo " $0 stop (stop JReport Server)"
   exit 1
   ;;
   esac
   exit 0
   ```
2. Create a soft link to `/etc/rc2.d/S99jrserver`.

   `ln -s /etc/init.d/jrserver /etc/rc2.d/S99jrserver`
3. Create a soft link to `/etc/rc0.d/K99jrserver`.

   `ln -s /etc/init.d/jrserver /etc/rc0.d/K99jrserver`

Running as a service on Linux

Running JReport Server as an OS service on Linux is more or less the same as with running on Unix. Here it is assumed that your default start up rc is rc5.

Setting up XVFB

1. Install XVFB.
2. Write a script `/etc/init.d/xvfb` as follows, and make it executable.
   ```
   #!/bin/sh
   mode=$1
   case "$mode" in
   'start')
   echo "start xvfb "
   if [ -f /usr/X11R6/bin/Xvfb ]
   then
   /usr/X11R6/bin/Xvfb :1 -screen 0 1152x900x8 &
   fi
   ;;
   *)
   echo " Usage: "
   echo " $0 start (start XVFB)"
   echo " $0 stop (stop XVFB not support)"
   exit 1
   esac
   exit 0
   ```
3. Create a soft link to `/etc/rc5.d/S97xvfb`.

   `ln -s /etc/init.d/xvfb /etc/rc5.d/S97xvfb`

Using rc to run JReport Server as a service

Assume that JReport Server has been installed to `/JReport/Server`.

1. Write a script `/JReport/Server/bin/JRServer` as shown below, and make it executable. Here it is assumed that JReport Server is running on a machine with IP address 127.0.0.1.
   ```
   #!/bin/sh
   DISPLAY=127.0.0.1:1.0
   export DISPLAY
   /JReport/Server/bin/JRServer -silent "$@"
   ```
2. Write a script `/etc/init.d/jrserver` as follows, and make it executable.
   ```
   #!/bin/sh
   mode=$1
   if [ ! -d /JReport/Server ]
   then # JReport not installed
   exit 1
   fi
   case "$mode" in
   'start')
   if [ -d /JReport/Server ]
   then
   echo "Starting JReport Server"
   cd /JReport/Server/bin/;
   JRServer -silent &
   fi
   ;;
   'stop')
   if [ -d /JReport/Server ]
   then
   echo "Stopping JReport Server"
   /JReport/Server/bin/CmdSender localshutdown &
   fi
   ;;
   *)
   echo " Usage: "
   echo " $0 start (start JReport Server)"
   echo " $0 stop (stop JReport Server)"
   exit 1
   ;;
   esac
   exit 0
   ```
3. Create a soft link to `/etc/rc5.d/S98jrserver`.

   `ln -s /etc/init.d/jrserver /etc/rc5.d/S98jrserver`
4. Create a soft link to `/etc/rc5.d/K98jrserver`.

   `ln -s /etc/init.d/jrserver /etc/rc5.d/K98jrserver`

If all has been carried out successfully, the installation of the service will now have finished. JReport Server is now ready to run as a daemon process.

Running within an application server

In addition to running as a standalone server and as a service, JReport Server can also run as a servlet inside a Java application server. Since JReport Server is implemented using servlets and JSPs, it can work with any servlet-enabled web application server by assembling and deploying JReport Server as a Web Application Archive (WAR) or Enterprise Application Archive (EAR).

There is a separate chapter about integration with different application servers - Integrating JReport Server with a Java Application Server. It provides a general method for creating the deployable archive and detailed procedures for deploying the archive into some popular application servers.

Basic Concepts

This chapter gives you some basic JReport Server concepts. You can first go through this chapter to gain a general understanding about JReport Server before you use it. Also, while you are using JReport Server, if you do not understand any of the basic concepts, you can refer to this chapter to get help.

This chapter discusses the following JReport Server main concepts:

- Background tasks
- Scheduling
- Resource
- Version
- Security
- Integration
- JReport Server Cluster

Background tasks

JReport Server provides a background running system, which shows the status information of tasks submitted using the Run, Advanced Run, or Background Run mode. Status information includes: report name, report path and name, catalog path and name, running format, time when the task is started/ completed, time elapsed since the task is performed, and the status of the task. It allows you to view detailed information in a timely fashion.

The records saved in the background running system are cleared under the following conditions:

- JReport Server is restarted.
- The maximum time limit specified for the report result life has been reached. By default it is 86400 seconds (24 hours).
- The maximum time limit specified for the interval between a user logout and login has been reached. By default it is 300 seconds.
- If the number of records exceeds the number specified for the background task list (by default it is 100 records), the latest 100 records will be retained.

Scheduling

JReport Server provides a scheduling system which you can customize to suit your requirements. You can submit a scheduled task from web page and URL or by calling the Server API methods. However, before you can do this, you must first specify the report, catalog, task type and its launch type. In addition, you can also customize notification messages to notify others of whether or not the task is executed successfully.

User Task

In order to provide the means to run tasks defined outside of JReport on JReport Server, and to just use JReport Server's schedule function, JReport provides a task named User Task. With this task, you can implement a customized task with the schedule properties. Also, you can submit the user task from a web page, or by calling JReport Server API methods.

After creating a class that implements the UserTask interface in the jet.server.api package and adding the class to the class path, you can then submit the task either from a server web page or by calling Server API methods. The task can then be run by the server at the scheduled times just as if it were a report.

Trigger

The scheduling mechanism supports trigger conditions in addition to time conditions. Triggers are managed by name in JReport Server, so each trigger must have a unique name. After creating a trigger, you can submit a task that is bound with the trigger, and then fire the trigger to activate the task at anytime.

Trigger conditions are based on event driven modes. The server does not care whether a customized condition is ready. It only waits there for trigger firing events. Therefore, you determine whether the condition is ready before firing a trigger.

Triggers can also work together with time conditions for activating a scheduled task.

Related topics:

- Scheduling reports

Resource

JReport Server provides a resource system for managing a group of archive versions that can be processed or organized.

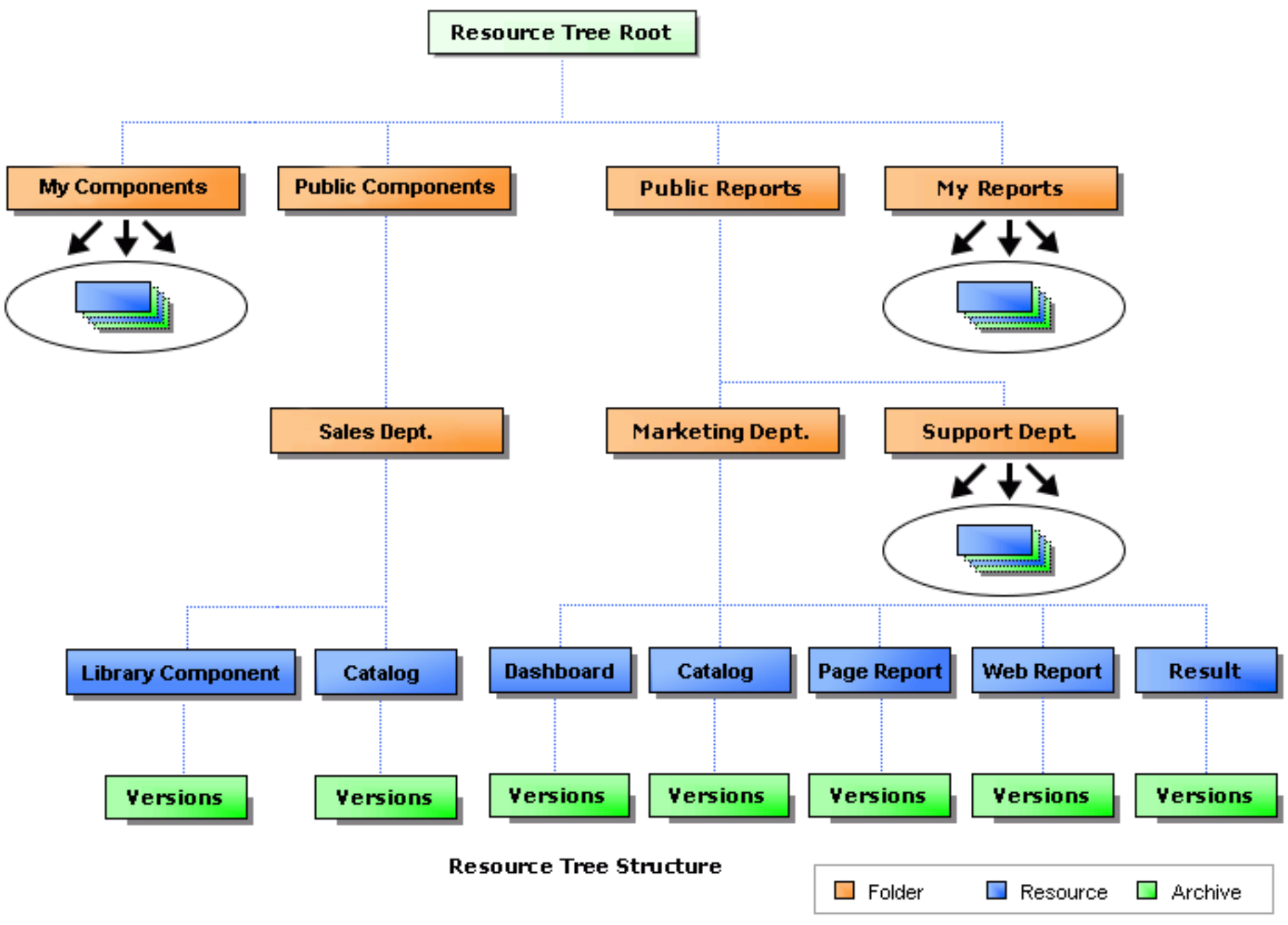
What is a resource

Generally, a resource refers to report or dashboard related material. To be exact, a resource in the JReport Server reporting system is a conceptual node. There are different types of resources, such as catalogs, reports, dashboards, library components, and their results. A resource can only hold versions of the same type.

Resource tree

All the resources are organized in a folder-tree structure. JReport Server defines an XML file called admin.xml, and the resource tree conforms to this file. This file is maintained automatically by JReport Server.

For example, your company has two departments - Support and Marketing. Each department has its reports on their own machine. There are some report documents that are submitted by the departments located on the machine where JReport Server runs. Now, suppose you are the administrator, and you would like to organize these files and folders into the Resource Tree. The following diagram may help you to figure out the framework on which you should build the resource tree.



Resource Tree Structure

The resource tree consists of the following three layers:

- **Folder layer:** Basic resource tree element that builds the main framework for the resource tree. There are four built-in folders in the root of the resource tree - My Components, Public Components, My Reports, and Public Reports. A folder can be mapped to a real file path.
- **Resource layer:** An abstract layer, based on the Folder layer that hosts various types of archive versions and provides user access to the versions.
- **Archive layer:** A concrete layer, where the archive versions reside for executable resources, which function as the leaves of the resource tree.

Public Reports and My Reports

Public Reports and My Report are two built-in folders in the resource tree root for storing resources such as reports, dashboards and Visual Analysis templates. You can create your own folders in either of them. The Public Reports folder and the My Reports folder cannot be deleted.

The Public Reports folder contains public resources, and can be accessed by everyone.

The My Reports folder is a personal folder. It contains personal resources. Each user has one personal folder, specified by the administrator when the user account is created. The My Reports folder can only be accessed by its owner, and the user has full control over his/her personal folder. This folder is the default output location for resources run by the user.

Reports

There are two types of reports in JReport: page reports and web reports. A page report is a collection of report tabs and each report tab can have multiple pages, while a web report has no report tabs and is always displayed as a web layout report with just one page.

JReport Server supports viewing, advanced running and scheduling of reports. The background run/scheduled/active/completed record are based on report level.

On JReport Server, you can also create page reports with Page Report Studio and web reports with Web Report Studio.

Dashboards

Dashboards, created using JDashboard, allow you to see the big picture by comparing charts, tables, and other components side-by-side.

Visual Analysis templates

Visual Analysis templates are the data status saved via Visual Analysis.

Public Components and My Components

Public Components and My Components are two built-in folders in the resource tree root for storing library components. Their behaviors resemble the Public Reports and My Reports folders.

The Public Components folder contains public components and can be accessed by everyone. The My Components folder is a personal folder that contains personal components for each dashboard user.

Library components

Library components are used to build dashboards. They are able to present data via intuitive components such as charts, crosstabs, tables, and geographic maps. Library components are created and edited using JReport Designer, and then are published to the component library on JReport Server for use in dashboards.

Related topics:

- Managing resources

Version

JReport Server provides a versioning system for controlling the resources contained in the resource tree. To understand what the versioning system is, first you have to understand the resource mechanism in JReport Server. A resource in the JReport reporting system is a conceptual node, which holds a group of archive versions that can be processed or organized in JReport Server. Information of these versions is stored in the System DB database that JReport Server uses, while version files are saved in the directory - `<reporthome>\history`.

What is a version

All the server resources in the resource tree are controlled by versions. A version is the fundamental unit of the resource tree, and your resources might change over time. JReport Server uses a versioning system to create and manage resources that have changed in content and properties owing to updates issued upon them.

All the resources in the resource tree have versions. A large portion of resource management tasks are done by managing resource versions.

Different version types

The versions in JReport Server fall into the following major categories:

- **Catalog Version**
  The version of a catalog file.
- **Report Version**
  The version of a report file.
- **Report Result Version**
  The version of a report result file.

  The report result can be generated and maintained in two places - the resource tree and the built-in version folder.

  When you advanced run or schedule a report to publish to the versioning system, you can choose an archive location to generate the report result. You can generate the report result in the built-in version folder, the My Reports folder or the Public Reports folder in the resource tree.

  The report results generated in the resource tree are standalone results and can have their own versions, while those generated in the built-in version folder can only be bound with their respective reports.
- **Dashboard Version**
  The version of a dashboard file.
- **Library Component Version**
  The version of a library component file.

Real path of versions

If you check the property of a version, you will find its real path. Remember that version information is stored to a database, and version files are stored in the directory `<reporthome>\history`. For the report InvoiceReport.cls, the report version's real path is `<reporthome>\history\1\JReport_System_User894485281\InvoiceReport.cls`, which is the actual report result path on disk and stored in the server database. That is, when you click the InvoiceReport.cls report result resource on the server interface, you are accessing it on the disk, only the path to it is stored in the database. And this works the same for the other types of versions.

Archive policy

JReport Server uses an archive policy to control the resource versions. You can control whether or not to use multiple versions for a specific resource. Also, you can define the maximum number of versions that can be listed in the version table.

The archive policy can be applied to a single resource individually, or to many resources in a folder as a whole.

Related topics:

- Managing versions

Security

JReport Server provides a security system for setting up and maintaining security on it, allowing you to protect your resources from inappropriate access by other users.

To help you understand security in detail, the following security features with their concepts are described below:

Realm

A realm is an abstract security concept, which hosts the resources and authentication entities on JReport Server. There can be more than one realm on the server and each realm is independent from others. The resources and authentication entities that reside in different realms are different.

At runtime, only one realm can be active and only the users and resources in the active realm are accessible. A realm is identified by a unique name, which can contain any characters other than forward slash (/) and backward slash (\).

The authentication entities consist of user accounts, group accounts and role accounts.

User

To use JReport Server, you must have a user account, which consists of a unique user name and password. JReport Server verifies your identity when you type your user name and password and then logs you on. If your user account has been disabled or deleted, JReport Server prevents you from accessing the services that JReport Server provides, in order to ensure that only valid users have access.

JReport Server comes with two built-in user accounts, which are admin and guest. The built-in user accounts cannot be deleted. The admin user account can neither be deleted nor disabled.

Group

The principal group, which represents an organization of user accounts, is available for managing users. Users or groups can then be added into a group as its child members, and therefore inherit the resource and folder permissions from the group.

Role

Users must have certain user rights and permissions in order to perform tasks on resources. Roles, which represent an aggregate of permissions, help you to efficiently assign the appropriate user rights and permissions to users. Assigning roles to users gives them the user rights and permissions that they require to perform their jobs with. A role can also be assigned to other groups or roles, and thus groups or roles can inherit the permissions of other roles.

JReport Server comes with two built-in role accounts, which are administrators and everyone. The built-in role accounts cannot be deleted. The administrator role account can neither be deleted nor disabled.

Permission

Permissions, associated with resources and folders which locate in the Public Reports or Public Components folder, are the rules that are granted to users to control their access to resources and folders.

Permissions in JReport Server include:

Permission	Description
Visible	Allows or denies viewing object names in the resource tree or version table, such as folders, resources, and archive versions.
Read	Allows or denies viewing object properties, versions, and, if it is a folder, folder contents.
Write	Allows or denies publishing folders and resources, changing the properties (not including permission settings) of the objects in the resource tree or version table, such as folders, resources, and archive versions, and modifying version table settings.
Execute	Allows or denies running reports in normal and Advanced mode (for report type resources only) or running dashboards in the view mode (for dashboard type resources only) or opening Visual Analysis templates in Visual Analysis (for analysis type resources only).
Edit	Allows or denies running dashboards in the edit mode (for dashboard type resources only).
Schedule	Allows or denies submitting resources to schedules (for report type resources only).
Delete	Allows or denies deleting objects in the resource tree or version table, such as folders, resources, and archive versions.
Grant	Allows or denies granting permissions to other users, groups or roles. Users, groups or roles that have obtained the Grant permission are also endowed with the other permissions, and can grant these permissions except the Grant permission itself.
Update Status	Allows or denies updating report status, and if it is a folder, the status of reports in the folder.

Privilege

Privilege is a mode which manages permissions. It can be used to manage different access permissions unrelated with nodes. Privilege of JReport Server manages the following access permissions for users:

- **Publish**
  The privilege of publishing resources to JReport Server.
- **Advanced properties**
  The privilege of viewing advanced information of version properties such as catalog connections and report related resources.

Alias

JReport Server organizes files and directories into a Resource Tree. Aliases are used to provide different "views" of the tree for different users. For example, you may set an alias resource tree (based on the resource tree) for Tanya, so that she can only see the marketing resource node and can directly enter into the report folder she is interested in. An alias is a combination of users and resource nodes.

Organization

JReport Server administrator can organize users into different organizations. An organization is a group of users, groups, and roles that has its own administrator. The organization administrator can define which users can use which dynamic connections within the organization.

Related topics:

- JReport Security System
- Managing security

Integration

JReport Server can be seamlessly integrated with any other Java application server to meet the information delivery needs of a single department or an entire enterprise. It contains a rich set of APIs that allow for seamless integration and is implemented using Java Servlet technology and Java Server Page (JSP). These servlets and JSP pages enable the user to work with any Java EE compliant application server that supports a Servlet Container and administer the JReport Server remotely through a web browser.

Related topics:

- Integrating JReport Server with a Java Application Server

JReport Server Cluster

A JReport Server Cluster is a distributed cluster in which a group of servers work together to provide cluster-wide shared resources, security, schedules and version services. In a JReport Server Cluster, all clustered servers play exactly the same role and any one can exit from the cluster any time.

The JReport Server Cluster provides the following major benefits:

- **Manageability**: All users and resources can be controlled from a clustered server, remotely.
- **High-Availability**: When one server fails to perform, the tasks running on it will be re-allocated to other servers. If a server has already been fully utilized, the tasks sent to it will be allocated to the other servers.
- **Scaleable**: You can add or remove servers dynamically according to your needs.

There are many nodes (clustered servers) in a JReport Server Cluster. Every clustered server has the same responsibility. You can set a clustered server to perform a specific role in a JReport Server Cluster by configuring its properties.

Related topics:

- JReport Server Cluster

Accessing JReport Server

You can access JReport Server through a web browser such as Internet Explorer, Firefox, or Google Chrome.

Starting and logging onto JReport Server

To log onto JReport Server, first start the server via one of the following ways:

- Double-click the **JReport Server 13.5 Update 2** shortcut on your desktop.
- Click **Start** > **All Programs** > **JReport 13.5 Update 2** > **Start Server**.
- Run the JRServer.bat/JRServer.sh file located in `<install_root>\bin`.
- Run the startup file from a command prompt or shell. For example, assume that JReport Server has been installed in `C:\JReport\Server` on Windows or `/opt/JReport/Server` on Linux, you can type the following commands:

```
C:\>cd JReport\Server\bin
C:\JReport\Server\bin>JRServer.bat
```

```
$cd /opt/JReport/Server/bin
$./JRServer.sh
```

Then,

- **To access the JReport Console page:**
  1. Open a web browser and set the URL to `http://ip_or_hostname:port` (by default, the port for accessing the JReport Console page is 8888).
  2. On the welcome page, type your user name and password as assigned by your administrator. For first time users, the default user name and password are admin.
  3. Click **Login** and the JReport Console page will be displayed.
- **To access the JReport Administration page:**
  1. Click **Start** > **All Programs** > **JReport 13.5 Update 2** > **System Admin**, or open a web browser and set the URL to `http://ip_or_hostname:port` (by default, the port for accessing the JReport Administration page is 8889).
  2. In the Sign in dialog, type your user name and password as assigned by the administrator. For first time users, the default user name and password are admin.
  3. Click **Login** and the JReport Administration page will be displayed.

Tip: If you don't know the IP address of the machine on which the server runs, and if it is the same machine where you run JReport Server, you can use *localhost* instead of the IP address. You can also open a console window such as telnet on the server machine and type **hostname**, then the name of the host will be displayed.

Fast launch pad for local users

Local users can also access JReport Server in a fast way with the launch pad, which is a convenient entry to access the server without having to start it. The launch pad provides some key functions of JReport Server by setting up the connections to corresponding JReport Server JSPs, which are:

- Creating dashboards, reports and Visual Analysis
- Viewing the JReport sample reports
- Scheduling to run reports by time or event
- Managing security principals
- Configuring server profiles to customize the server interface and functionality
- Viewing the product demos

To access the launch pad, click **Start** > **All Programs** > **JReport 13.5 Update 2** > **Launch Pad**.

Additional login channel for admin users

JReport Server provides a special channel that automatically creates an extra user session for management purposes if the license limit of the maximum number of concurrent users has been reached. The extra user session cannot be used to run reports or submit schedules. It can only be used by admin users for performing management operations. If your JReport Server license has a bounded limit to the maximum number of concurrent users, this feature will take effect.

Only one extra valid user session can be created and used within this special channel at any time. If an extra user session has already been created and it is still valid, your request for login will be prompted with a confirmation page asking you whether or not to close the existing extra user session. If you select Yes, a new extra user session will be created for you to perform management operations. Otherwise, you will not be allowed to log onto JReport Server.

Note: If all normal user sessions have been used up, a request from a client viewer will be denied permission to log onto JReport Server. Only a normal user session can use a client viewer to log onto JReport Server, not the extra user session.

Logging off and shutting down JReport Server

To log off JReport Server, click the **Logout** link on the upper right corner of the JReport Administration/ Console page.

To shut down JReport Server normally:

- In a standalone environment, click the **Shut Down the Server** button on the JReport Administration page, or click **Start** > **All Programs** > **JReport 13.5 Update 2** > **Stop Server**.
- In an integrated environment, shut down the application server according to the vendor's instructions.

Also, JReport provides a feature for handling an abnormal system exit that enables the program to close itself gracefully when the Java virtual machine (JVM) is terminated in response to a user interrupt, such as typing ^C, or a system-wide event such as user logoff or system shutdown.

SSL in standalone JReport Server

JReport Server supports HTTPS requests in standalone mode. Secure ports for HTTPS requests should use different ports from non-secure ports for HTTP requests. By default, port 6888 and 6889 are set as the secure ports separately for accessing JReport Console UI and JReport Administration UI. The URL for visiting JReport Server via HTTPS schema is like this:

```
https://IP_address or localhost:6888
```

SSL support is disabled by default. You need to enable it and configure corresponding settings in order to use HTTPS schema to visit JReport Server UI. This can be done either on the JReport Administration page or in the server.properties file located in the `<install_root>\bin` directory.

To enable SSL function via the administration UI:

1. Log onto the JReport Administration page. The default port is 8889.
2. Click **Configuration** on the system toolbar, and then select **Service** from the drop-down menu.
3. Check the **Enable Secure Socket Layer Connection** option.
4. Make sure that Secure Port and Secure Administration Port use different port numbers from Port and Administration Port.
5. Specify the other settings about Keystore.
6. Click **Save**.
7. Restart JReport Server in order for the settings to take effect.

To enable SSL function in the server.properties file:

1. Open the server.properties file located in the `<install_root>\bin` directory.
2. Set httpserver.ssl.enable to **true**.
3. Set the other properties starting with httpserver.ssl to meet your requirements.
4. Save the server.properties file.
5. Restart JReport Server in order for the settings to take effect.

Notes:

- JReport does not provide a keystore file since Jinfonet is not a trusted certificate authority and just provides a Keystore File Path option for you to configure the location of your trusted keystore file. There are many trusted authorities that can provide keystore files. Here is an example of creating a keystore file: http://docs.oracle.com/cd/E19509-01/820-3503/ggsxx/index.html.
- JReport Server Monitor does not support SSL.

Working with Reports

After successfully logging onto the JReport Console page using the user name and password assigned by your administrator via a web browser, you can then perform tasks according to your requirements. For example, you can view reports in different formats and schedule tasks for the reports.

Pick a task from the following:

- Running reports
- Scheduling reports
- Specifying parameter values
- Using JSPs to print page reports

Related topics:

- Managing resources
- Managing versions
- Managing tasks

Tip: You can customize the JReport Console > Resources page to suit your requirements by setting your own preferences. To do this, click **Tools** > **Preferences** on the task bar of the Resources page, then specify the settings in the Preferences dialog as required. For example, you can specify the default viewing format when directly run a report from this page, set which columns will be shown in the resource information table, and predefine the properties for each export format which will be applied when you run or schedule a report on this page.

Running reports

If you know the default format for viewing reports and this format is what you expect, use direct running. If you would like a different view format or to customize some other information such as parameter values, the running priority, how to save the result, the expiration time, and so on, use advanced running.

Direct running

Direct running reports is to use the default format setting to view the report result. For page reports (for example, .cls) the default format is Page Report and they will run in Page Report Studio. For web reports (.wls) they will be opened in Web Report Studio by default.

If the reports are created on JReport Server, the default format is controlled by the Default Format for Viewing Report option in the Profile > Customize Server Preferences > General tab. When this option is set to Page Report or Applet, web reports will run in Web Report Studio by default.

If the reports are created using JReport Designer, the default format is controlled by the Default Format for Viewing Report property in the Report Inspector. If this property is set to <Server Setting>, the viewing format will be determined by the above setting on server. For page reports this property is available to each report tab in the reports, while for web reports it is available to each report.

Since a page report can contain multiple report tabs and these report tabs may have different default format settings, for example, report tab 1 is Page Report, tab 2 HTML and tab 3 Excel. In this case, there will be different direct running results according to which report tab is the last-time focused tab in the page report when the page report is saved at report design time. If it is report tab 1, all the tabs in the page report will be opened in Page Report Studio. If it is report tab 2 or 3, only tab 2 or 3 will be displayed in the specified format, and this applies to all the other formats except Page Report.

To directly run a report, on the JReport Console > Resources page, browse to the report, then do one of the following:

- Click the name of the report in the Name column of the Resources page.
- Select the report row and click **Run** > **Run** on the task bar of the Resources page.
- Select the report row, right-click in the row and select **Run** from the shortcut menu.
- Put the mouse pointer over the report row and click the **Run** button on the floating toolbar.

Then, if the report contains parameters, the Enter Parameter Values dialog will be displayed. Specify the parameter values according to your requirements.

Advanced running

1. On the JReport Console > Resources page, browse to the report you want to run.
2. Do either of the following to open the Advanced Run dialog.
   - Select the report row, then on the task bar of the Resources page, click **Run** > **Advanced Run**.
   - Select the report row, right-click in the row and select **Advanced Run** from the shortcut menu.
   - Put the mouse pointer over the report row and click the **Advanced Run** button on the floating toolbar.
3. In the General tab, for a page report, select the report tabs in the page report you want to run. If there are multiple dynamic connections available, expand the **Select Dynamic Connection** section and select a connection from the drop-down list. Click **Connection Properties** to view the connection information if needed. Then, specify the other options as required.
4. In the Parameter tab, specify the parameter values if the report has parameters.
5. In the Format tab, choose a format to view the report result, and set the other settings.
6. In the Archive tab, archive the report result version according to your requirements.
7. If you want to limit the amount of time that the report is allowed to run, in the Duration tab, specify a time duration for the task, and ask JReport Server to cancel the task or to notify you or someone else of the task status via e-mail if the task has not yet finished running when the task duration is up. For detailed information, see Task-level timeout for advanced run and schedule tasks.
8. Click **Finish** to view the report in the format you specified.

   When running a page report in Page Report Studio, the report processing page will appear, on which you can choose to cancel the running of the report, or to make the report run in background mode. Click **Cancel** on this page if you decide to cancel, **Background** if you want the report to run in background mode, or just wait for processing to complete. If you cancel the report from running, you can choose whether to cancel the running query used by the report in the database at the same time by configuring the JdbcDriversConfig.properties file. For details, refer to Canceling running query.

Notes:

- If you choose to view a page report in Page Report format, besides the selected report tab, all the other tabs in the report will also be run.
- When you run a report in Advanced mode in HTML format, the names of page navigation links in the report, such as First, Previous, Next, and Last, can be localized according to your requirements. For details, refer to Localizing page navigation links in HTML report outputs.
- When you run a report in Advanced mode in Applet format or in HTML format with Applet Chart is checked for it using Google Chrome, the report can run correctly only when NPAPI is enabled for the web browser. To do this, you need to type **chrome://flags/** in the address bar of the web browser, then scroll down and set Enable NPAPI Mac, Windows to **Enable**.
- By default, the Duration tab is not displayed in the Advanced Run dialog. To make it available, the Enable Task Duration option on the JReport Administration page > Configuration > Advanced panel must be checked.

Running a page report in background mode

When you run a report in Page Report Studio, if the report contains a large amount of data, you need to wait several minutes before the report result is displayed, and during this period, you have to remain on the report processing page, or choose to cancel the run. Now JReport enables you to switch running page reports to background mode.

To make a report running in Page Report Studio to run in background mode, on the report processing page, click the **Background** button.

You can also specify to run page reports automatically in background mode after a specified time period by setting preferences. To do this:

1. On the JReport Administration page, click **Profile** on the system toolbar and then select **Customize Profile** from the drop-down menu. Or on the JReport Console page, click **Profile** on the system toolbar, then click **Customize Profile** on the task bar of the Profile page.
2. Click the **Page Report Studio > Properties** > **Advanced** tab.
3. Check **Background Mode Timeout** and specify the time allowed for a page report to run in foreground mode.
4. Save the settings, then when a report runs in Page Report Studio and the results have not yet been generated after the specified time, it will be automatically switched to run in background mode.

Reports running in background mode are listed in the Background Tasks table of the My Tasks page, which shows detailed running information of the reports, such as report path and name, catalog path and name, running format, time when the task is started/completed, and so on. Also, you can control the status of the reports running in background mode according to your requirements. For example, you can choose to delete, stop, or restart tasks (for details, see Managing tasks in the task tables).

When a report completes running in background, you can open it from the Background Tasks table, and once the report is opened, the task will automatically be removed from the table.

Scheduling reports

JReport Server can run reports at a specified time or periodically by scheduling tasks for reports. The scheduled tasks will be recorded by the server according to their different executing status.

Pick a topic from the following for details about how to schedule tasks in JReport Server:

- Scheduling reports with dialog
- Scheduling a task containing a bursting report
- Scheduling a customized task using User Task
- Recording scheduled tasks
- Viewing scheduled report result
- Importing and exporting scheduled tasks
- Adding TaskListener

Notes:

- When you schedule to publish a report to Page Report Result format, if the report is linked to another report, in the Page Report Result file, the link will no longer be supported, and if you schedule to publish the report to several formats and Page Report Result format is included at the same time, the link will not be available in the other format outputs either.
- When you schedule to publish a report to HTML format, the names of page navigation links in the report, such as First, Previous, Next, and Last, can be localized according to your requirements. For details, refer to Localizing the page navigation links in HTML report outputs.

Scheduling reports with dialog

You can schedule tasks for a specified report with dialog as follows:

1. Do either of the following to open the Schedule dialog.
   - On the JReport Console > Resources page, browse to the report you want to schedule to run, then:
     - Select the report row, then on the task bar of the Resources page, click **Run** > **Schedule**.
     - Select the report row, right-click in the row and select **Schedule** from the shortcut menu.
     - Put the mouse pointer over the report row and click the **Schedule** button on the floating toolbar.
   - On the JReport Console page, click **My Tasks** on the system toolbar, then select the **Scheduled** tab. Click **New Schedule** on the task bar of the My Tasks page, then in the New Schedule dialog, check the **Select Report to Create Schedule** radio button, specify the folder and then the catalog in which the report you want to schedule to run is saved, then select the report and click **OK**.
2. In the General tab, specify the name for the task in the Schedule Name text box. For a page report, you need to select the report tabs you want to run from the page report. You can choose multiple normal report tabs or one bursting report at a time (for scheduling a bursting report, see Scheduling a task containing a bursting report). When multiple normal report tabs are selected, check **Export to One File** if you want to export them to just one result file. If there are multiple dynamic connections available, expand the **Select Dynamic Connection** section and select a connection from the drop-down list. Click **Connection Properties** to view the connection information if needed. Then, specify the other options as required.
3. In the Parameter tab, specify the parameter values if the report has parameters.
4. In the Publish tab, specify the type of the task.

   Six task types are provided by JReport Server: To Version, To Disk, To E-mail, To Printer, To Fax and To FTP. Choose the type you want to publish, and then set the settings for the specified type.
5. In the Conditions tab, specify the time for when the task is to be performed in the Time sub tab, and select or create a trigger to bind with the task in the Trigger sub tab.
6. In the Notification tab, specify to notify someone via e-mail of when the task is finished and whether it is successful or unsuccessful.
7. In the Duration tab, specify a time duration for the task, and ask JReport Server to cancel the task or to notify you or someone else of the task status via e-mail if the task has not yet finished running when the task duration is up. For detailed information, see Task-level timeout for advanced run and schedule tasks.
8. Click **Finish**, and JReport Server will then perform the task.

Notes:

- By default, the Duration tab is not displayed in the Schedule dialog. To make it available, the Enable Task Duration option on the JReport Administration page > Configuration > Advanced panel must be checked.
- When scheduling to publish a page report to the versioning system or to disk, the Page Report Result and JReport Result formats are based on the report level, that is, the report with all selected report tabs will be output to a single file. As for the other formats, each selected report tab will be output to a separate file.

The following are some specific scheduling examples:

- Example 1: Publishing a report to the versioning system
- Example 2: Publishing a report to disk
- Example 3: Publishing a report to e-mail
- Example 4: Publishing a report to printer
- Example 5: Publishing a report to fax
- Example 6: Publishing a report to an FTP site

Example 1: Publishing a report to the versioning system

In this example, a task is set up and will be performed immediately. The generated result is asked to be kept for 30 days.

1. On the JReport Console > Resources page, select the report row, right-click in the row and select **Schedule** from the shortcut menu to display the Schedule dialog.
2. In the General tab,
   a. Specify a name for the task in the Schedule Name text box.
   b. For a page report that contains multiple report tabs, you need to select the report tabs you want to run from the page report in the Select Report Tabs section. Change the reports tabs order using the **Move Up** or **Move Down** button. Check the **Export to One File** option if you want to export the selected report tabs to one result file.
   c. If there are multiple dynamic connections available, expand the **Select Dynamic Connection** section and select a connection from the drop-down list. Click **Connection Properties** to view the connection information if needed.
   d. Expand the **Report Information** section, click **Select Another Catalog** to specify another catalog for the report if required.
   e. Select the report version and catalog version from the corresponding drop-down lists.
   f. Assign a priority to the task from the Priority drop-down list.
   g. Expand the **Advanced** section, if the report has a style group, check the **Enable Style Group** checkbox and choose the style group.
   h. Check the **Enable Converting Encoding** option if required and specify the encoding before and after converting from the corresponding drop-down lists.
   i. If **Enable NLS** is available, you can select it and then specify a language to display the report contents with.
   j. Define the encoding for the report by selecting from the drop-down list.
   k. Check **Use the default DB user and password defined in catalog**.
   l. Check the **Add TaskListener to be Invoked** option, and input the class name (for details, see Adding TaskListener).
   m. Check the **Enable Auto Recover Task** option to specify auto recovering settings.
3. In the Parameter tab, specify the parameter values as required if the report has parameters.
4. In the Publish tab,
   a. Click the **To Version** sub tab, then check **Publish to Versioning System**.
   b. Select the required formats and set the format settings.
   c. Check the **Built-in Version Folder** option in Archive Location to save the report result version in the built-in version folder.
   d. Set **0** for the Maximum Number of Versions.
   e. Check the **Result Auto-delete** option and define the result to expire in 30 days.
5. In the Conditions tab, select the **Time** sub tab, define the time zone from the Time Zone drop-down list, then from the Time Type drop-down list, choose **Run this task immediately**.
6. If you want to notify someone of when the task is finished by sending an e-mail, go to the Notification tab and then set the settings.
7. If you want to specify a timeout for the task, specify the settings in the Duration tab as required.
8. Click **Finish** to have the task performed.

Then, click **My Tasks** on the system toolbar. While the task is being performed, you can see a record of it in the Running tab. On completion it will be put into the Completed tab.

Notes:

- When publishing a report to XML format, if you want to use the URL to get the .xsd file, follow the steps below:
  1. Put the existing XML schema file in `<intall_root>\public_html`.
  2. Input `http://IP address:8888/name of the existing XML schema file`.
- There is another way to publish the report result to version. If you have set the property server.version.from.temp to true in the server.properties file in `<install_root>\bin`, or selected the option Enable "Publish to Versioning System" for Background Tasks View on the JReport Administration page > Configuration > Advanced panel, you will get the link Publish to Version System on the system toolbar of the JReport Console page. Click the link to publish the result to version.

Example 2: Publishing a report to disk

In this example, you will learn how to set up a task to publish the report result in various file formats to the file system repeatedly at the start of each month.

1. Take steps 1, 2, 3 as in Example 1.
2. In the Publish tab of the Schedule dialog, click the **To Disk** sub tab, select the required format, specify the result location, and set the format settings according to your requirements.
3. In the Conditions tab,
   a. In the Time sub tab, define the time zone from the Time Zone drop-down list, then from the Time Type drop-down list, choose **Run this task periodically**.
   b. In the Duration box, specify a time period for when the task will be performed.
   c. Select **Monthly** from the Date drop-down list and keep the default to run the first day of every 1 month.
   d. Keep the Time settings as default.
   e. If you also want to set an event which needs to occur before running the report, select a trigger to bind with the task from the Select a trigger to bind drop-down list in the Trigger sub tab, then specify the trigger logic with time condition.

      For detailed information about the logic between the trigger condition and the time condition, see Trigger tab.
4. If you want to notify someone of when the task is finished by sending an e-mail, go to the Notification tab and set the settings.
5. If you want to specify a timeout for when the scheduled report will be allowed to run, specify the settings in the Duration tab as required.
6. Click **Finish** to have the task performed.

Then, click **My Tasks** on the system toolbar, you will see that the scheduled task has been recorded in the Scheduled tab. Since you have not specified the duration *Run until a time* for this task, it will not stop being performed until you delete or disable it from the Scheduled tab.

Notes:

- When you specify to publish the report result to the server resource tree, if the specified folder has a real path, the result will be put to the real path. Otherwise it will be put to the default disk location where server resources are.
- If you specify to publish the report result to a non-existent folder on disk, JReport Server will automatically create it.
- If you use a fixed name to perform the task periodically, only the most recently generated report result will be kept. In order to keep the report result generated every time, you should use the dynamic result file name. Similarly, you are able to input the dynamic directory path to avoid report management difficulties, since when a fixed directory path is specified, too many results may be generated in one directory. For more information, see Appendix 5: Dynamic names.

Example 3: Publishing a report to e-mail

In this example, you will learn how to set up a task to publish the report result to e-mail.

1. Take steps 1, 2, 3 as in Example 1.

2. In the Publish tab of the Schedule dialog, click the **To E-mail** sub tab, then from the Mail To list, select to whom the report result will be sent. If required, click the **Edit** button to edit the specified e-mail.

   If you want to create another e-mail, click the **New** button, then fill in every field, select the format in which you want to export the report result and set the settings according to your requirements. When you choose to specify a report result as an attachment to e-mail, you need to specify a file name for the attachment. For details about settings of creating a new e-mail, refer to Schedule dialog - To E-mail.

3. In the Conditions tab, select the **Time** sub tab, define the time zone from the Time Zone drop-down list, then from the Time Type drop-down list, choose **Run this task immediately**.

4. If you want to notify someone of when the task is finished by sending an e-mail, go to the Notification tab and set the settings.

5. If you want to specify a timeout for the task, specify the settings in the Duration tab as required.

6. Click **Finish** to have the task performed.

Then, click **My Tasks** on the system toolbar. When the task is being performed, you can see a record of it in the Running tab and on completion it will be put into the Completed tab.

Example 4: Publishing a report to printer

In this example, you will learn how to set up a task to publish the report result to a printer.

1. Take steps 1, 2, 3 as in Example 1.
2. In the Publish tab of the Schedule dialog,
   a. Select the **To Printer** sub tab and then check **Publish to Printer**.
   b. Select a JDK print method for the report result in the Select Print Method field.
   c. Type a name with the path of the printer in the Printer field.
3. In the Conditions tab, select the **Time** sub tab, define the time zone from the Time Zone drop-down list, then from the Time Type drop-down list of the Time tab, choose **Run this task immediately**.
4. If you want to notify someone of when the task is finished by sending an e-mail, go to the Notification tab and set the settings.
5. If you want to specify a timeout for the task, specify the settings in the Duration tab as required.
6. Click **Finish** to have the task performed.

Then, click **My Tasks** on the system toolbar. When the task is being performed, you can see a record of it in the Running tab and on completion it will be put into the Completed tab.

Note: When there is no printer connected with JReport Server, and you schedule to publish a report to a printer, the server may crash or throw an exception.

Example 5: Publishing a report to fax

In this example, you will learn how to set up a task to publish the report result to fax. Before you can fax the report result, you must first have your modem configured. Otherwise a warning message will be displayed when you try to export to fax.

1. Take steps 1, 2, 3 as in Example 1.
2. In the Publish tab of the Schedule dialog, click the **To Fax** sub tab, check the **Publish to Fax** option and then fill in every field and set the settings according to your requirements.
3. In the Conditions tab, select the **Time** sub tab, define the time zone from the Time Zone drop-down list, then from the Time Type drop-down list, choose **Run this task immediately**.
4. If you want to notify someone of when the task is finished by sending an e-mail, go to the Notification tab and set the settings.
5. If you want to specify a timeout for the task, specify the settings in the Duration tab as required.
6. Click **Finish** to have the task performed.

Then, click **My Tasks** on the system toolbar. When the task is being performed, you can see a record of it in the Running tab and on completion it will be put into the Completed tab.

Example 6: Publishing a report to an FTP site

In this example, you will learn how to set up a task to publish the report result to an FTP site.

1. Take steps 1, 2, 3 as in Example 1.
2. In the Publish tab of the Schedule dialog,
   a. Click the **To FTP** sub tab, then click the **New** button to set up a new FTP site or click the **Edit** button to edit a specified FTP site in the FTP To list.
   b. Fill in every field, select the format in which you want to send the report result and then set the settings according to your requirements.
3. In the Conditions tab, select the **Time** sub tab, define the time zone from the Time Zone drop-down list, then from the Time Type drop-down list, choose **Run this task immediately**.
4. If you want to notify someone of when the task is finished by sending an e-mail, go to the Notification tab and set the settings.
5. If you want to specify a timeout for the task, specify the settings in the Duration tab as required.
6. Click **Finish** to have the task performed.

Then, click **My Tasks** on the system toolbar. When the task is being performed, you can see a record of it in the Running tab and on completion it will be put into the Completed tab.

Note: Dynamic result name for FTP task is currently not supported.

Scheduling a task containing a bursting report

In a large enterprise reporting deployment, it is important to handle both large amounts of data as well as a large number of users. Report bursting enables running a report once and distributing the report result to multiple recipients who each will receive a subset of the report result.

Bursting reports can be distributed to e-mail or FTP addresses, to disk, to the JReport versioning system, or to the security system members such as users, groups, and roles.

End users can submit a schedule task which contains only one bursting report to JReport Server. When a bursting task is activated, it will create a main bursting task and some sub bursting tasks. The system will guarantee bursting tasks compete with normal tasks for system resources. The bursting tasks can be given lower priority if desired (set queue.policy to 1).

- **Main bursting task**: It is responsible for getting/splitting data and distributing work to the sub tasks. There can be only one main bursting task for a sub bursting task.
- **Sub bursting task**: It is responsible for generating the report result according to split data and sending the result to the address of the bursting recipient.

On JReport Server, direct running and advanced running actions are supported for normal reports but not bursting reports. A report containing only bursting report tabs cannot be run directly or in Advanced mode, it must be scheduled.

Scheduling is supported for both types of reports excluding the combination of the two types: for normal reports, multiple reports can be scheduled at a time; however for bursting reports, only one can be scheduled. For a scheduled bursting task, seven kinds of result file formats are supported: HTML, PDF, Excel, Text, RTF, XML, and PostScript. In addition, when scheduling to run a bursting report, you can make it generate not only the bursting result by applying bursting schemas but also the non-bursting result based on whole data without data splitting.

Scheduling a bursting report to generate bursting result

Though a bursting report may have one or more bursting schemas, you need apply one or more of them in order to get a bursting result. To do this, select a bursting report and schemas, and then specify required parameter values in the General tab of the Schedule dialog. Then a tab named Bursting Result is displayed in the Publish tab and only the corresponding sub tabs that are defined in the selected bursting schemas' recipients in JReport Designer are available. For example, a bursting report has three bursting schemas: Schema 1 defines recipient E-mail and Disk, Schema 2 defines recipient FTP, and Schema 3 defines recipient JReport Server Version. If Schema 1 and Schema 3 are selected, only To E-mail, To Disk, and To Version sub tabs will be shown in the Bursting Result tab of the Publish tab for the bursting result.

Once you choose to schedule to run a bursting report, you should at least specify the schema you want to apply to the selected bursting report or select Non-bursting result in order to submit the task.

The following list tells which tab will be displayed in the Publish > Bursting Result tab of the Schedule dialog for which recipient address specified in bursting schema.

Recipient	Sub tab in the Publish tab
E-mail	To E-mail
FTP	To FTP
Disk	To Disk
JReport Server Version	To Version
JReport Server User/Group/Role - User E-mail	To E-mail
JReport Server User/Group/Role - User Private Folder	To Version

When scheduling a bursting report, you needn't specify the destination in the Publish tab since the recipient addresses have been included in the bursting schema. However, you are allowed to give a file name to the subset of report result instead of using the default name.

Default name for bursting result files

Sometimes you may not want to specify a file name for each bursting result when defining recipients. The bursting system will give it a name as generated by the system.

The default name format is: ReportName + "_" + BurstingKey + suffix (result format type). When there are multiple bursting key columns, connect each one by the character "_".

Converting to String

When a bursting key is of one of the following data types, it will be converted into String so as to make a valid result file name:

- Integer, Float, Character: Same as Java, these data types are transferred to string directly.
- Date and Time: All data and time formats will be transferred to a date format: yyyy-MM-dd hh:mm:ss.
- Currency: Currency will be transferred to the number without the currency mark ($ or others).

Name length

In the JReport Server resource system, the resource name only supports up to 64-character length. If a bursting result file name is longer than that, the system will trim it down automatically.

In order to avoid using the same name in the same path, an index will be appended to the result name, for example: report1_USA_Maryland1.pdf, report1_USA_Maryland2.pdf.

Notes:

- All bursting sub results will apply the security information of the bursting task submitter.
- When running a page report containing both normal and bursting report tabs:

  For direct running:

  When running it to Page Report format, only the normal report tabs are opened.

  When running it to other formats, if the default focused report tab is a normal report, it will be opened directly; if the default focused report tab is a bursting report which cannot be run, a warning message will be displayed asking the end user to select a normal report tab to run using advanced run.

  For advanced running:

  You can only choose the normal report tabs to run.

Scheduling a bursting report to generate non-bursting result

Besides generating bursting result for a bursting report, you can also generate non-bursting result for the report without applying any bursting definition, which is based on full data without data splitting.

To generate non-bursting result, select the **Non-bursting result** option in the General tab of the Schedule dialog. Then a tab named Non-bursting Result appears in the Publish tab, and all these sub tabs - To Version, To Disk, To E-mail, To Printer, To Fax, and To FTP - are available in this tab for the non-bursting result.

Scheduling a customized task using User Task

In order to meet the requested requirement to run tasks defined outside of JReport on JReport Server, and to just use JReport Server's schedule function, JReport has provided a task named User Task. With this task, you can implement a customized task with the schedule properties. You can also submit the user task from a web page, or by calling the JReport Server API methods.

To schedule a customized task using User Task:

1. Create a task class that implements the UserTask interface and add the path of the class file to the ADDCLASSPATH variable in setenv.bat, which is located in `<install_root>\bin`. You can find the interface in the jet.server.api package available in `<install_root>\help\api`. JReport provides a demo class APIDemoDynamicExportTask.java in `<install_root>\help\samples\APIServer` for your reference.

2. Create a task properties file defining the formats of exporting the task. For example, the content of the properties file is:

   ```
   jrs.rst=result_rtf
   jrs.pdf=true
   jrs.text=true
   jrs.excel=true
   jrs.ps=true
   jrs.html=result_html
   ```

3. Submit the task either from a server web page or by calling Server API methods. The task can then be run by the server.

   - **To submit the customized task from a server web page:**

     a. On the JReport Console > Resources page, browse to the desired report, put the mouse pointer over the report row and click the **Schedule** button on the floating toolbar. The Schedule dialog is then displayed.

     b. Specify the settings in the General, Publish, Conditions, and Duration tabs as required. Here the Publish tab settings should be switched to those of User Task by clicking the link on the right bottom of the tab. Then, the name of the task class file you have defined that implements the UserTask interface and the task properties that define the export formats must be provided. You can either input the task properties manually or import them from the task properties file that you have created.

     c. Upon finishing, click **Finish** to submit the task.

   - **To submit the customized task by API methods:**

     a. Set the user task class name to the property value of APIConst.TAG_TASK_CLASS.

     b. Define the user task properties with the property APIConst.TAG_USER_TASK_PROP with the formats as follows: jrs.user_task_prop= jrs.rst=result_rtf&jrs.pdf=true&jrs.text=true&jrs.excel=true&jrs.ps=true&jrs.html=result_html. Since the jrs.user_task_prop is used to transfer multiple user task properties, the values set to this property must be formatted to be separated with "&" character.

     c. Set a display name for the class with the property APIConst.TAG_USER_TASK_DISPLAY_NAME.

     d. Except for the above properties, define the other schedule properties as you do with a default task, see APIDemoPublishRpt.java in `<install_root>\help\samples\APIServer` for reference. Then following the API demo you can submit a customized task on the server.

Click **My Tasks** on the system toolbar. When the task is being performed, you can see a record of it in the Running tab and on completion it will be put into the Completed tab.

Note: You can either schedule to use the Default Task or the User Task at one time. If you specify to schedule a report as a default task, you will not be able to schedule it as a user task, and vice versa.

Recording scheduled tasks

JReport Server records every scheduled task. Click **My Tasks** on the system toolbar of the JReport Console page, then the following tabs will be displayed: Scheduled, Running, and Completed. The tasks are recorded in different tables according to their executing status. A task is placed in the Scheduled table as soon as it is submitted by the user. It will go to the Running table when it is running, and when it is finished by JReport Server, it will then go to the Completed table.

Related topics:

- Managing tasks

Viewing scheduled report results

When a scheduled task is finished, you can view the result of the scheduled report as required. To view the result, first of all you need to know the corresponding result's location. To get the location information:

1. On the JReport Console page, click **My Tasks** on the system toolbar, then click the **Completed** tab, where all the successfully scheduled tasks are recorded.
2. Locate the task in the tab and click the name of the task in the Schedule Name column.
3. In the Result Details table, the location information of the scheduled result is available in the Details column.

Viewing results scheduled to version

To view report results that are scheduled to version, you can choose whether to view via the JReport Console page or via URL.

Viewing via the JReport Console page

One way to view results that are scheduled to version on the JReport Console page is via the scheduled task records. To do this:

1. Click **My Tasks** on the system toolbar, then click the **Completed** tab.
2. In the tab, locate the right task and click the name of the task in the Schedule Name column.
3. In the Result Details table, the links to different result formats are available for viewing in the To Version row.

And another way to achieve the same purpose is via the server resource tree which to some extent varies with the archive location type specified in the Publish > To Version tab:

- If the archive location has been set to Built-in Version Folder:
  1. On the JReport Console > Resources page, browse to the row that the original report is in.
  2. Do either of the following:
     - Select the report row and click **Tools** > **Version** on the task bar of the Resources page.
     - Select the report row, right-click in the row and select **Version** from the shortcut menu.
     - Put the mouse pointer over the report row and click the **Version** button on the floating toolbar.
  3. In the Report Result Versions tab, the scheduled results of different format types are listed in the Result column. Click the format links or action buttons to view the results.
- If the archive location has been set to My Reports Folder or Public Reports Folder, which requires providing a path and a name for the scheduled result in the server resource tree:
  1. On the JReport Console > Resources page, browse to the row that the result is in.
  2. Put the mouse pointer over the result row and click the **Version** button on the floating toolbar (or you can use one of the other two methods shown in the above procedure to display the version table).
  3. In the Result Versions tab, the scheduled results of different format types are listed in the Result column. Click the format links or action buttons to view the results.

Tip: The unviewed version results are highlighted in bold. You can cancel the highlighting by setting the property web.version.mark_unviewed to false in the server.properties file in `<install_root>\bin`.

Viewing via URL

For the report results that are scheduled to version, you can also view them via URL. One important thing in the URL method is that you need to know the file name of the result that you want to view. To get this information, you can take the steps explained at the beginning of the document.

Using command jrs.view_ver_rst

See the example:

```
http://localhost:8888/jrserver/SampleReports/SampleReports.cat/Employee Information.cls?
jrs.cmd=jrs.view_ver_rst&
jrs.hist_file=1%5cJReport_System_User327406359%5cEmployee Information.rst&jrs.
result_type=1.
```

See details about the usage of the command jrs.view_ver_rst.

Using viewVersion.jsp

For example:

```
http://localhost:8888/jinfonet/viewVersion.jsp?jrs.cmd=jrs.view_ver_def&jrs.
rst_version=2&jrs.ver_suff=.html&
jrs.report=%2fEmployee Information.cls&type=rstfile&jrs.path=%2fEmployee Information.cls
```

Notes:

- To view RSD version results using JSP, you should use dhtml.jsp instead of viewVersion.jsp. For example:

  ```
  http://localhost:8888/dhtmljsp/dhtml.jsp?jrs.rst_version=10&jrs.report=/Banded_Link.
  cls&type=rstfile&jrs.path=/Banded_Link.cls
  ```

- When viewing a scheduled page report result with cached report bursting via URL, you should add `jrs.is_pls_result=true` in the URL, for example: `http://localhost:8888/dhtmljsp/dhtml.jsp?jrs.rst_version=1&jrs.is_pls_result=true&jrs.report=/PLS.cls&type=drstfile&jrs.path=/PLS.cls`

Viewing results scheduled to disk

When scheduling a report to disk,

- If you choose Publish to Server Disk Path, you are required to provide a disk file path and file name with correct format type as the suffix for each report tab in the report. After scheduling succeeds, you can find the corresponding result files available at the specified location on the computer where JReport Server is installed.
- If you choose Publish to Server Resource Tree, you are required to provide a path following the server resource tree and file name with correct format type as the suffix for each report tab in the report. For example,
  - To follow the My Reports folder path, start with "/USERFOLDERPATH/admin/".

    Example: /USERFOLDERPATH/admin/report1.pdf
  - To follow the Public Reports folder, start with "/".

    Example: /report2.html.

  If the specified folder which is the parent folder of the result file has a real path, the generated result file will be saved to the real path; if the folder doesn't have a real path, the generated result will be saved to `<server_install_root>/jreports/`, which is the mapped disk path of the root node "/" in the specified path.

Viewing results scheduled to e-mail/printer/fax/FTP

When a report is scheduled to e-mail, printer, fax, or FTP, you can view the scheduled result if the specified address or location is available to you.

Importing and exporting scheduled tasks

In JReport Server, you can export a scheduled task to a script file which will then be saved on your own disk as a script file. In addition, you can import a script file from the disk file to generate a scheduled task.

To export a scheduled task to a script file and save it in the disk:

1. On the JReport Console page, click **My Tasks** on the system toolbar.
2. In the Scheduled tab, select the rows that one or more scheduled tasks are in.
3. Click **Tools > Export to Script** on the task bar of the My Tasks page (if only one task is selected, you can also right-click in the task row and select **Export to Script** from the shortcut menu, or put the mouse pointer over the task row and click the **Export to Script** button on the floating toolbar), then modify the script text in the Edit Script box as required.
4. Click **OK** to export the specified scheduled task to a script file.
5. Specify the directory and name for this script file in the File download dialog.

To import a script file from your disk:

1. On the JReport Console page, click **My Tasks** on the system toolbar, then select the **Scheduled** tab.
2. Click **New Schedule** on the task bar of the My Tasks page.
3. In the New Schedule dialog, check the option **Import Script to Create Schedule**.
4. Click the **Browse** button to select a script file from your disk file, then click **OK** to import the specified script file and modify the script text in the Edit Script box as required.
5. Click **OK** to generate a scheduled task.

Note: If you just updated from an older version of JReport Server, there may be some old scripts saved in your server. In order to use these old scripts, you can click the **Import old script from server** link in the Import Script page to select an old script to import it to generate a scheduled task. To use this link, you must logon JReport Server as an administrator role.

Adding TaskListener

When viewing or scheduling a report, JReport Server enables you to call your Java application before or after the process.

In JReport Server, a TaskListener interface has been provided in the package jet.server.api for receiving task events before or after running. You can specify one Java class to implement this interface for a task event. When the event of this task occurs, the corresponding methods in the listener will be invoked. The interface contains two methods: beforeRun and afterRun, enabling you to set your Java application call before or after the process of viewing a report or setting up a schedule. Your applications will return true or false. For true, JReport Server will go on running. While for false, JReport Server will stop there.

Below is an example illustrating how to add TaskListener when setting up a schedule on a report.

1. Develop your Java class to implement the interface. Here TestTaskListener.java is used, which is available in `<install_root>\help\samples\APITaskListener`.
2. Compile TestTaskListener.java to generate the class file.
3. Edit the batch file setenv.bat in `<install_root>\bin`. Assuming that TestTaskListener has been saved in `C:\JReport\Server\tasklistener`, add the path of the class file (`C:\JReport\Server\tasklistener`) to the ADDCLASSPATH variable in setenv.bat.
4. Start JReport Server and set up a schedule on a report, check **Add TaskListerner to be Invoked** in the General tab of the Schedule dialog, then input the class name. In this example, input TastListener, and then submit the task.
5. In this example, the class returns True.

   Print out the task and schedule properties before and after running the task. You will then get task and schedule information in the command window before and after the task is run.

You can also define properties of your own and transmit them through ServerInfo. To do this, use APIConst.TAG_USERDEFINED_PROPERTY_PREFIX as the prefix for the properties.

For example, if you want to transmit the properties host_name, host_ip and hosp_protocol, you will need to insert the properties, before calling the method runTask, into the properties named prop, as follows:

```
prop.put(APIConst. TAG_USERDEFINED_PROPERTY_PREFIX+"host_name", "host");
prop.put(APIConst. TAG_USERDEFINED_PROPERTY_PREFIX+"host_ip", "127.0.0.1");
prop.put(APIConst. TAG_USERDEFINED_PROPERTY_PREFIX+"host_protocol"+ "TCP/IP");
```

You can get the value of the properties listed above through the server info object, serverInfo, in the method beforeRun or afterRun of the TaskListener class. See the example below:

```
host_name=serverInfo.getTaskProperties().get(APIConst.TAG_USERDEFINED_
PROPERTY_PREFIX+"host_name");
host_ip=serverInfo.getTaskProperties().get(APIConst.TAG_USERDEFINED_
PROPERTY_PREFIX+"host_ip");
host_protocol=serverInfo.getTaskProperties().get(APIConst.TAG_USERDEFINED_
PROPERTY_PREFIX+"host_protocol");
```

Note: All properties without the prefix APIConst.TAG_USERDEFINED_PROPERTY_PREFIX will be denied and discarded by JReport Server.

Specifying parameter values

When running or scheduling reports with parameters, you may need to specify values for the parameters. The way to specify a parameter value varies with the type and properties of the parameter. Here are several ways you can use to specify parameter values:

- In the parameter value combo box, input the value manually or select the required one from the drop-down list. If you have chosen to automatically save parameter values, the saved value groups will be available on the top of the parameters' value lists for selection.
- Select or unselect the checkbox to specify a Yes/No value.
- Click the button ... to specify multiple values in the Enter Values dialog.
- Click the calendar button to specify a date and time value using either calendar or expression in the Calendar dialog. If you use an expression to specify the value, when you hover the mouse pointer over this value, a tip will appear showing its expression. Click on the value, the expression will be displayed in the value text box and you can edit the expression in the text box directly if you want. After editing, when you click elsewhere outside of the value text box:
  - If the edited expression is correct, a new value calculated by the expression will be displayed in the parameter value text box.
  - If the edited expression is wrong, no value will be created and the expression itself will be displayed and highlighted in red in the value text box. Correct the expression, or click the calendar button to specify a value.
- Click the **Use Saved Values** button and select a previously saved parameter value group to apply to the report. For details, see Manually saving parameter values.

Note: You are recommended not to use blank as the thousands separator in Number-typed parameter values under French locale, otherwise your input will not be correctly recognized because of a JVM bug. For details, see http://bugs.sun.com/view_bug.do;jsessionid=c8cdaf911b20ffffffffd9fc6340b30d670?bug_id=4510618.

This document also introduces some useful techniques for setting parameter values.

Saving parameter values for reuse

When specifying parameter values for reports, you may want to save the specified parameter values for reuse next time. JReport provides two ways of saving. One is users decide when and which parameter values to save, the other is JReport saves each applied or submitted parameter values automatically. When the saved number in either way reaches the maximum, the oldest record will be removed. The number is calculated on a user-report basis. Take a report with two parameters for an example, supposing the maximum number is set to 3. Each user can save at most three groups of parameter values for the report. Each group contains the values of the two parameters.

To switch on the function of saving parameter values, go to the Profile > Customize Server Preferences > Advanced tab, select **Yes** to the option Enable Saving Parameter Values, select a way to do the saving: Manually or Automatically, and then specify a maximum number to limit the saved value groups which is called the auto complete parameters list for each user-report pair.

Manually saving parameter values

When you have chosen to manually save parameter values, the Use Saved Values button will be available on the upper right corner of the parameter page. By clicking this button, you will get the following:

- **Value List**
  The value list contains all the parameter value groups you have saved for the report. Select a group to apply all the values in the group to the corresponding parameters. You can also remove any of the unwanted value group by selecting it from the list and then clicking next to the value list.
- **Save Values**
  Saves the current displayed parameter values as a group for reuse afterwards. To do this, type a name for the group in the text box which shows "Save current values into list" and then click . The saved group will then be added into the value list above. Once it reaches the maximum number, the latest group will replace the oldest.

Using automatically saved parameter values

When you have chosen to automatically save parameter values, each time a user submits a group of parameter values to a report, the group is saved by JReport automatically. The next time the same user runs the report, the auto saved parameter values will be available in the parameters' value lists for selection on the parameter page.

Customizing default parameter values

When specifying parameter values for a report, if you would like the current specified parameter values to be saved as the default values for the parameters next time when you run or schedule the report, select the **Save as default** option on the parameter page (to make this option available, you need to make sure the Enable Setting Default Parameter Values For option is selected for the corresponding resource in the Profile > Customize Server Preferences > Advanced tab).

If you want the saved default parameter values to be applied directly to the report next time when you run it without popping up the Enter Parameter Values dialog, select the option **Do not show the screen again** in the dialog (to make this option available, you need to make sure the Enable Hiding Initial Parameter Dialog For option is selected for the corresponding resource in the Profile > Customize Server Preferences > Advanced tab). However, if the last-time saved default values cannot completely match the current report parameters, the dialog will still be displayed. This option only hides the parameter dialog popped from operation on the console. It does not affect the case of running report using URL way.

In addition, JReport provides you with the Parameter Settings dialog which allows you to pre-customize the default parameter values for a report before running it. This dialog can also be used to show the Enter Parameter Values dialog again after it has been hidden. To do this, check **Re-enable Parameter Screen** in the dialog.

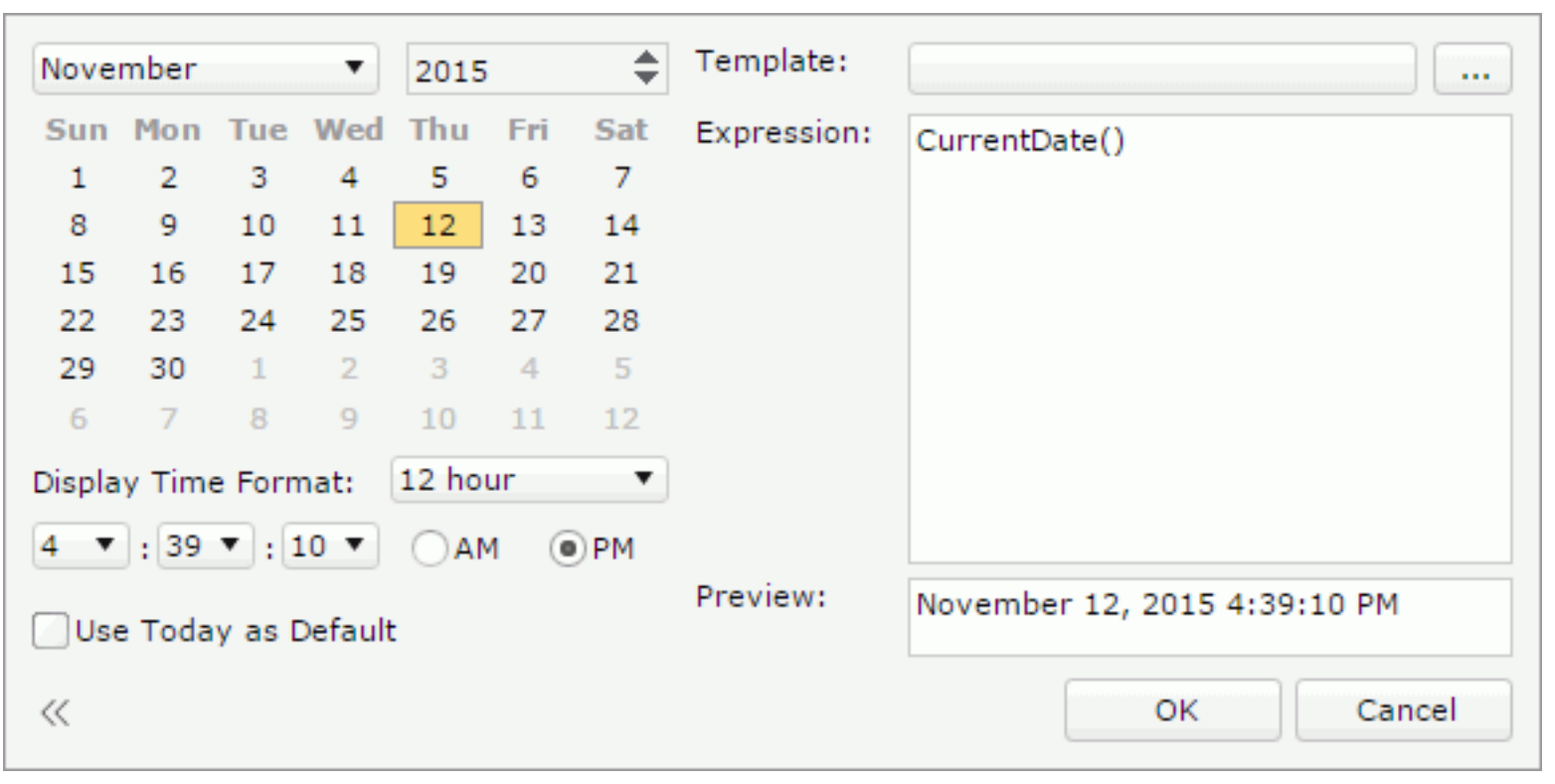
The above options are user-report level settings, that is to say, they take effect when both the same user and report are matched. This also applies to admin users, and therefore admin cannot customize the settings for all users.

Using built-in functions to set date and time parameter values

For a parameter of the Date, DateTime, or Time type, you can customize a dynamic date and time value by creating an expression using built-in functions.

To do this:

1. In the places where you are able to specify a value for the parameter, you may see the calendar button . Click this button and the Calendar dialog appears.



2. The Template drop-down list on the right provides some predefined expressions for you to use. You can customize your own expression either based on an existing template or by directly editing the contents in the Expression text box.

   If you cannot see the Template label on the right, click **>>** on the bottom left corner.
3. JReport's built-in Date/Time formula functions are available for inserting into your expression. To make use of them, click ... to display the functions.
4. The Preview box calculates the result of the current expression. Each time you modify the expression, you can click in the box to refresh the result. If it is an invalid expression, an error message will be shown in the box and no change will be made to the calendar and the currently selected date/time will be remained in the calendar.
5. When done, click **OK** to add the value.

When running reports via URLs, you can also use expressions to specify date and time parameter values. The format is Parameter_Name={"exp":"Complete_Expression_String", "SelectedDate":"'MM/dd/yyyy"}. If the expression does not contain the function SelectedDate(), there is no need to add this part - "SelectedDate":"'MM/dd/yyyy" and the format is Parameter_Name={"exp":"Complete_Expression_String"}. For example, to specify p_EndDate=today(), write it as p_EndDate={"exp":"today()"}. The following is an example of running the CustomerAnalysis.cls:

```
http://localhost:8888/jinfonet/runReport.jsp?jrs.cmd=jrs.web_vw&jrs.catalog=%
2fSampleReports%2fSampleReports.cat&jrs.report=%2fSampleReports%2fCustomerAnalysis.
cls&jrs.param$P_StartDate=01/01/2010&jrs.param$p_EndDate={"exp":"today()"}&jrs.
result_type=8
```

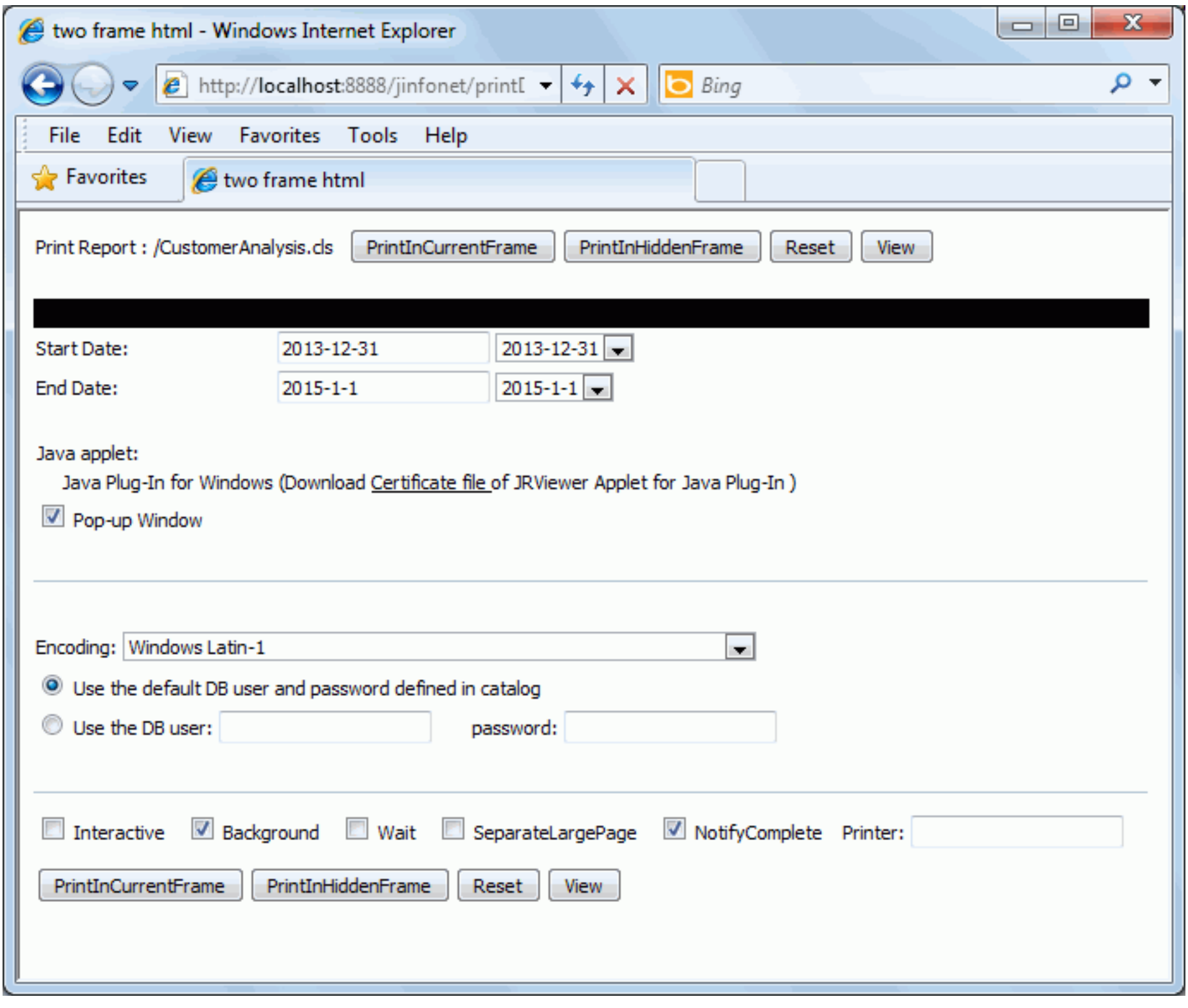
Using JSPs to print page reports

JReport Server provides the following demo JSPs which enable you to print a page report without any view in the client side, and they are available in `<install_root>\help\samples\JSPSamples\PrintReport`.

- **printDemo.jsp**
  Provides frames to load printCustomerlist.jsp and printReport.jsp.
- **printCustomerlist.jsp**
  Shows how to set parameters for printing the demo report CustomerAnalysis.cls by using the ViewerApplet. This JSP calls printReport.jsp. If you want to print the other reports, you can follow this JSP file as an example to write your own JSP. In your own JSP, you should modify the values of "cat" and "rptName".

  ```
  String cat = "/SampleReports/SampleReports.cat"; //request.getParameter(APIConst.TAG_CATALOG);
  String rptName = "/SampleReports/CustomerAnalysis.cls"; //request.getParameter(APIConst.TAG_REPORT);
  ```
- **printReport.jsp**
  Shows how to print reports by using the ViewerApplet. The example JSP printCustomerlist.jsp calls this JSP.

Before running the JSPs, copy them to `<intall_root>\public_html\jinfonet`. Then, start JReport Server and access printDemo.jsp using the URL `http://localhost:8888/jinfonet/printDemo.jsp`. The following page appears:



PrintInCurrentFrame

Specifies to call printReport.jsp and load applet of Page Report Studio Frame.

PrintInHiddenFrame

Specifies to call printReport.jsp without loading applet of Page Report Studio Frame.

Reset

Resets the previous options.

View

Views this report.

Interactive

If checked, you could specify the print setup in the Print dialog.

Background

If checked, the print job will run in the background.

UseJDK1.1

If checked, you will use instance PrintJob of JDK11 to print the report.

Wait

If checked, you have to wait until the print job is finished.

SeparateLargePage

If checked, the large page will be separated into several pages automatically.

NotifyComplete

This parameter is used with the parameter wait. After the print job is finished, a box will pop up to note you, and you could do any other work instead of keep on waiting.

Printer

Specifies the printer to implement the print job.

Page Report Studio

With JReport Server, you can obtain report result in different formats, such as HTML, PDF, and Excel. For the HTML format, there are two viewing modes - pure HTML or Dynamic HTML. Page Report Studio provides a dynamic report view at the client side. You can change options which enable the results to be displayed to your requirements.

Interactive information empowers you to slice and dice your business data, to dynamically change your view of data, and to analyze the data to glean useful business information. In short, interactive information enables you to customize your view of business information. With JReport, any report can be made interactive, extending the "life" of the report by allowing you to easily sort, navigate, and filter data via Page Report Studio. This wide range of functionality, including the ability to drill down on data, enables you to quickly derive value from your business intelligence data.

Page Report Studio provides support for many features. You can dynamically modify - filter, search, sort and drill - reports to obtain unique and personal data views. The Page Report Studio toolbar and interactive web objects (images, buttons, text fields, checkboxes, radio buttons, drop-down lists and so on) can be embedded into reports or JSPs at design time, enabling you to control, customize and navigate report views. Page Report Studio also supports a web design feature, allowing you to create reports using report oriented data structures, save your report, and even save your custom modifications to existing reports.

This chapter covers the following topics to help you better understand how Page Report Studio makes reports interactive and how you will benefit:

- Page Report Studio window elements
- General operations
- Ad hoc reporting
- Analytic reporting
- Working with reports via URL
- Opening multiple reports in one session
- Tuning Page Report Studio performance

Page Report Studio window elements

The main page of Page Report Studio consists of the user information bar, menus, toolbar, page report bar, Toolbox, Resource View panel, TOC Browser, and report area. The options for browsing or controlling a DHTML report are as follows:

Toolbar / Menu	Button	Tool Name	Description
File		New Page Report Tab	Creates a new report tab to the current page report based on an existing business view.
		New Page Report	Creates a new page report containing a report tab based on an existing business view.
		Open	Opens the Open Report Tabs dialog for you to open/close report tabs in current report.
		Rename Report Tab	Opens the Rename Report Tab dialog to give the open report tab a new name.
		Close Report Tab	Closes the current report tab if there is more than one report tab open in the report; or prompts you to close the report if there is only one report tab.
		Delete Report Tab	Deletes the current report tab if there is more than one report tab open in the report. This command is disabled when the last page of the current report tab does not display if Format Page on Demand in the Profile > Customize Profile > Page Report Studio > Properties > Advanced tab is selected.
		Save	Saves the report as a report version.
		Save As	Saves a copy of the report.
		Export	Exports the report result to disk or version in various formats.
		Page Setup	Shows the Page Properties dialog for you to specify the page layout settings for the report result.
		Printable Version	Shows the Printable Version dialog for you to print the current report result to a PDF/ HTML file.
		Exit	Closes the current report.
Edit		Undo	Undoes the last operation.
		Redo	Reverses the operation of Undo.
		Search	Shows the Search dialog for you to find specific text.
View		Toolbar	Shows or hides toolbars.
		User Information Bar	Shows or hides the User Information Bar, which displays the user name, catalog name and report name.
		Toolbox	Shows or hides the Toolbox panel which allows you to insert a component into the report.
		Resource View	Shows or hides the Resource View panel, with which you can add view elements to your report and create dynamic resources to use them in your report.
		TOC Browser	Shows or hides the TOC Browser, with which you can navigate the report data.
		Editing Marks	Shows or hides editing marks (dashed outlines for objects and report body). If the option is unselected, the editing mark will not be shown when a report object receives focus, and report objects cannot be moved or resized.
		Turn To	Provides a submenu for you to turn the report pages.
		Refresh	Runs the report using previously provided parameters. The Refresh operation fetches the data again.
		Zoom	Shows the Zoom dialog for you to set a zoom ratio for the report page.
		Options	Shows the Options dialog for you to set the skin and unit for Page Report Studio, and to customize toolbars.
		Show Grids	Shows grids in the report area.
		Snap to Grids	Snaps an object to grids when you move it by dragging and dropping in the report area. If this option is enabled, aligning objects will be made easier. To temporarily override the setting, press the ALT key as you move an object.
Insert		Label	Inserts a label into the report.
		Image	Inserts an image into the report.
		Banded Object	Inserts a banded object into the report.
		Table	Inserts a table into the report.
		Crosstab	Inserts a crosstab into the report.
		Chart	Inserts a chart into the report.
		Parameter Control	Inserts a parameter control into the report.
		Parameter Form Control	Inserts a parameter form control into the report.
		Filter Control	Inserts a filter control into the report.
		Navigation Control	Inserts a navigation control into the report.
		Special Fields	Inserts special fields into the report.
		Page Break	Inserts a page break into the report.
Report		Query Filter	Applies a filter to the business view used by certain component.
		Filter	Filters the report records according to the filter criteria you specify.
		Sort	Sorts the report records or groups in ascending or descending order on the fields you select.
		To Chart	Converts a crosstab into a chart.
		To Crosstab	Converts a chart into a crosstab.
		Rotate Table	Rotates a table to switch its appearance between the horizontal and vertical layout modes.
		Rotate Crosstab	Rotates a crosstab to exchange the columns and rows in the crosstab in order to create a different view of the crosstab.
		Merge	Merges selected tabular cells into one.
		Split	Splits a tabular cell into the specified number of rows and columns.
		Language	Allows you to specify the language in which to display the report. Available only when Enable NLS is checked in the Profile > Customize Server Preferences > Advanced panel on the JReport Console page.
		Max Records	Allows you to specify the maximum number of records retrieved by all components in the report.
		Use Dynamic Formula in Property	Allows you to apply dynamic formulas to control object properties.
		Style	Allows you to apply a style to the report.
		Change Parameters	Allows you to change the parameter values in the report.
Help		User's Guide	Opens Page Report Studio User's Guide.
		Jinfonet Software Home Page	Connects to Jinfonet Software Home Page.
		Technical Support	Accesses Jinfonet Technical Support.
		About Page Report Studio	Shows product information about Page Report Studio.
Standard Toolbar		New Report Tab	Creates a new report tab based on an existing business view.
		Open	Brings out the Open Report Tabs dialog for you to open/ close report tabs in current report.
		Save	Saves the report as a report version.
		Save As	Saves a copy of the report.
		Export	Exports the report result to disk or version in various formats.
		Printable Version	Shows the Printable Version dialog for you to print the current report result to a PDF/ HTML file.
		Undo	Undoes the last operation.
		Redo	Reverses the operation of Undo.
		Delete	Deletes the selected object.
View Toolbar		Toolbox	Shows the Toolbox panel for you to insert a component into the report. Click it again to hide the Toolbox.
		Resource View	Shows the Resource View panel, with which you can add view elements to your report and create dynamic resources to use them in your report. Click it again to hide the Resource View panel.
		Filter	Shows the Filter dialog, with which you can filter the report records according to the filter criteria you specify.
		Sort	Shows the Sort dialog, with which you can sort the report records or groups in ascending or descending order on the fields you select.
		Search	Shows the Search dialog for you to find specific text.
		Zoom	Enables you to enlarge or reduce the size of the report.
Analysis Toolbar		Rotate	Rotates a crosstab or rotates a table.
		Chart Type	Lists all available chart types for you to change the type of a selected chart.
	Style: JReportDemo	Style	Allows you to apply a style to the report.
Font format buttons	Arial 8	Font Face, Font Size	Changes the face and size of the selected font. Available only when a label or field is selected.
	B I U	Bold, Italic, Underlined	Makes the selected font in bold, italic or underlined style. Available only when a label or field is selected.
		Left, Center, Right	Makes the selected font left, center or right aligned. Available only when a label or field is selected.
Page navigation buttons	Max Records: All	Max Records	Allows you to specify the number of records retrieved by all components in the report.
	Page 3 of 4	Page Number	Displays the current page number. You can also input a page number in the page box and press Enter on the keyboard to go to that page.
		First	Goes to the first page of the current report tab.
		Previous	Goes to the previous page.
		Next	Goes to the next page.
		End	Goes to the last page.
Go To drop-down list	Go To: report1	Go To	Goes to the selected report tab or to the selected report tab level. If a report contains several reports tab, you can use this list to switch among the report tabs. Or, after you perform some going or drilling actions on a report tab, the structure of the report tab will be displayed in the list in a hierarchical view, with which you can return to any level of the report tab easily.
More Commands			When the Page Report Studio window is not maximized in Interactive View mode, the button will be displayed on the toolbar, by clicking which you can get all the other toolbar commands the small window hasn't enough space for.
	English (United States)	Language	Specifies the language in which to display the report. Available only when Enable NLS is checked in the Profile > Customize Server Preferences > Advanced panel on the JReport Console page.
Shortcut Menu		Filter	Provides submenu items for filtering the data in a banded object/table or removing the filtering.
		Sort	Provides submenu items for sorting records on the selected field in ascending/descending order, or removing the sorting.
		Drill Down	Drills data to a lower group according to predefined hierarchies.
		Drill To	Enables you to obtain a different view of data by switching among groups.
		Drill to By Value	Allows you to filter data based on groups while also obtaining a more detailed view of the data.
		Drill Up	Drills data to a higher group according to predefined hierarchies.
		Go To	Goes to any group to show its record information.
		Go Up	Goes up one group level to show the records of a higher-level group.
		Go Down	Goes down one group level to show the records of a child group.
		Go to Detail	Goes to the details of a group.
		Conditional Formatting	Enables you to add conditional format to the currently selected field.
		Search	Shows the Search dialog for you to search the report result for some text.
		Query Filter	Applies a filter to the business view used by the specified data component.
		Refresh	Re-fetches data of the specified data component.
		Properties	Shows a dialog for you to define the object's properties.

Notes:

- Page Report Studio has two view modes: Basic View and Interactive View, and the toolbar and menu commands that are available in each mode vary. Administrators can specify the default mode that will be applied when a report is opened in Page Report Studio in the Profile > Customize Server Preferences > General tab, and you can switch between the two modes by clicking the Basic View or Interactive View link on the Page Report Studio toolbar. However, you will not be able to switch the mode if the Show Link of Basic/Interactive View option in the Profile > Customize Profile > Page Report Studio > Properties > Default tab is unchecked.
- The options available on the Page Report Studio window are determined by the feature profile that is selected as the default profile in the Profile > Customize Profile > Page Report Studio > Features tab and the property setting on the Profile > Customize Profile > Page Report Studio > Properties tab (profile has the higher priority). The default Page Report Studio profile provides full options.
- The shortcut menu contents vary with the objects you right-click. The above table only lists some typical shortcut menu items. The following sections will guide you to use the shortcut menu for any object you may right-click.

General operations

After having opened a report in Page Report Studio, you can do the following general operations:

- **Managing report tabs**
  A page report can include one or more report tabs. The Go To drop-down list on the toolbar panel or the tabs across the top of the report lists the display names of all the open report tabs in the current report. Clicking the display name of an inactive report tab will make it active. You can manage report tabs in a page report easily as follows:
  - **Opening and closing a report tab**
    In a page report, a report tab can be shown or not. To close (hide) the active report tab, click **Menu** > **File** > **Close Report Tab** . If there are one or more report tabs open other than the active report tab, the close action will hide the active report tab; in the case that the active report tab is the only report tab open, the close action will prompt you whether or not to close the report.

    To open (show) a hidden report tab, click **Menu** > **File** > **Open** (or the **Open** button on the Standard toolbar) to display the Open Report Tabs dialog, in which the report tabs open in the current report are marked with a check symbol. Check the report tabs you want to open, uncheck the ones you want to close, and then click **OK**.
  - **Renaming a report tab**
    To rename a report tab, first activate it, then click **Menu** > **File** > **Rename Report Tab**. In the Rename Report Tab dialog, specify a new display name for the report tab.
  - **Deleting a report tab**
    To delete a report tab, first activate it, then click **Menu** > **File** > **Delete Report Tab**. The only report tab open cannot be deleted.

    Note: A JReport Live license for JReport Server is required in order to delete report tabs. If you do not have the license, contact your Jinfonet Software account manager to obtain it.

Tip: If the administrator has specified to switch report tabs using tabs in the Page Report Studio profile page, you can easily activate a report tab in a report by clicking the tab representing the report tab on the report tab bar, and closing, renaming and deleting a report tab can also be accomplished by right-clicking the report tab and choosing the corresponding command from the shortcut menu.

- **Turning the report pages**
  report tab includes more than one page, to turn between the report pages, you can:

- Click the **First Page** button , **Previous Page** button , **Next Page** button , or **Last Page** button on the View toolbar.

- Input a number into the page box Page 3 of 4 and press **Enter** to go to that page.

- Click **Menu** > **View** > **Turn To** and then click the corresponding command on the submenu. When he Page command is selected, the Turn To page dialog appears for you to input the page number.

- Use the scrollbar or mouse wheel to scroll up/down the report tab.

- **Navigating through the report data**
  can use the TOC Browser to navigate through a report tab. To show the TOC Browser, click **Menu** > **iew** > **TOC Browser**.

he TOC Browser, expand the Report node, select a component or a node with the group value that you nt to browse to. The page that contains the component or the matching data will then be shown.

e table of contents on the TOC Browser is organized into a tree structure. The root node represents the ort tab that you are currently viewing. The component names indicate components in the report tab. The up values show hierarchical groups.

report tabs designed in JReport Designer, the nodes displayed on their TOC tree as well as the format of TOC tree can be customized.

- **Refreshing the report result**
  fetch the data of the current report again, you can click **Menu** > **View** > **Refresh**.

- **Applying a style**
  tyle can be applied to a report in order to change its appearance and characteristics. You can create and up your own styles in JReport Designer. When you publish your reports to JReport Server, you can ude these custom styles with the published reports. When you run a report, the style feature will be abled and you can select a style to apply to the report.

- **Applying a style to a report**
  o apply a style to a report, make sure nothing is selected in the report, then click **Menu** > **Report** > **tyle** and select the required one from the submenu, or select the required style from the style drop-own list Style: JReportDemo on the toolbar. When a style is applied to the whole report, all omponents in the report will take a uniform appearance.

- **Applying a style to a data component**
  he data component (banded objects, crosstabs, charts, and tables) in a report can take a different style f its own. To apply a style to a data component, select it and then click **Menu** > **Report** > **Style** and elect the required one from the submenu, or select the required style from the style drop-down list tyle: JReportDemo on the toolbar, or right-click the component and select **Apply Style** from the hortcut menu to select the required style in the Apply Style dialog. If the data component is contained in banded object (for a chart, in a table also), you can also select the Inherit Style option to make it inherit he style of its parent data component.

wever, if there is only one style available to the report, this style will be applied to the report by default, which case, you will find that all these style related commands are hidden.

- **Changing the report parameter values**
  here are parameters in the current report, you can change the parameter values at runtime. To do this:

  1. Click **Menu** > **Report** > **Change Parameters** to display the Report Parameters dialog.
  2. Specify the value for each parameter.
  3. Click **OK** to run the report with the specified parameter values.

  can also use parameter web controls to dynamically change the parameter values of a report at time. For details, see Applying web controls.

- **Opening links**
  en a page report is developed with links in JReport Designer, which refer to locations specified by URLs, hail addresses or other reports, you can click the links' trigger objects in Page Report Studio to open the ts. However, if the click priority of the link action is not specified to be the highest at report design time, have to right-click on the trigger objects and click the corresponding link item on the shortcut menu to en the links. For example, if a trigger object is linked with a detail report, you need to right-click the ect and select Detail Report from the shortcut menu to open the detail report. Meanwhile, when opening detail report, you may be prompted with the Encoding dialog, asking you to provide encoding and DB urity information before the report result is produced. Click OK in the dialog if you want to run a detail ort using the same encoding and DB security settings as that of the master report.

  Note: When opening a link report or detail report, a "link path", which tracks the linking actions, will be displayed in the Go To drop-down list on the navigation bar if the link report or detail report is opened in the current window. Clicking an item in the list will switch to the corresponding report.

- **Printing the report result**
  can print the report result to a PDF or HTML file. To do this, click **Menu** > **File** > **Printable Version** (or **Printable Version** button on the Standard toolbar). In the Printable Version dialog, specify the tings as required and click **OK**. The PDF or HTML result file is then opened in an associated program with ich you can print the result to a printer.

- **Configuring Page Report Studio features**
  JReport Administration page provides default settings for you to use Page Report Studio features, and trols whether the settings on the JReport Console page can be configured. A user playing the ministrator" role may change the settings in the JReport Administration page so as to enable or disable e features. Then you can configure Page Report Studio preferences on the JReport Console page, that is, can decide whether or not to enable the features which have been enabled on the JReport ministration page. After you have made changes to Page Report Studio settings on the JReport Console ge and saved them, Page Report Studio features available for you will be consistent with your new tings on the JReport Console page. For details, see Customize Profile.

- **Setting Page Report Studio options**
  e Report Studio allows you to set the skin and customize toolbars. To do this:

  1. Click **Menu** > **View** > **Options** (or right-click anywhere on the toolbar area and select **Options** from the shortcut menu) to open the Options dialog.
  2. In the Option tab, set the skin of Page Report Studio user interface.
  3. In the Customize tab,
     - To modify a toolbar, select it in the Current Toolbar box, remove those unnecessary items from the Selected Tools box, and add required tools from the Available Tools box. Click or to adjust the order of the tools on the toolbar.
     - To add a toolbar, click to show the New Toolbar Name dialog, then specify the toolbar name, click **OK** to return to the Options dialog, and set the tools for the new toolbar.
     - To delete a toolbar, select it and click .
  4. To load the default settings, including the skin, and the three built-in toolbars, namely Standard, View, and Analysis, click the **Restore Defaults** button.
  5. Click **OK** to apply the settings.

  Tip: To close a toolbar, right-click anywhere on the toolbar area, then on the shortcut menu, select the item corresponding to the toolbar name. You can also do this to open an invisible toolbar, such as a newly-created one. The open/close toolbar operation can also be achieved by clicking the corresponding item on the Toolbar submenu of the View menu.

- **Tuning report page magnification**
  can zoom in or out the report page by selecting a magnification from the Zoom list on the View lbar. You can also click **Menu** > **View** > **Zoom** to show the Zoom dialog, and then specify the gnification.

- **Setting up the page**
  set up the report page, click **Menu** > **File** > **Page Setup**. In the Page Properties dialog, specify the page e, the orientation, and the margins as required.

- **Adding/Removing page breaks**
  ew page will be generated in a report tab only when the space in a page has been fully occupied or there sts a break control in this page, for example, a page break that is inserted in the page.

  insert a page break, click **Menu** > **Insert** > **Page Break**, then click the mouse button on the destination ere you want the page break to be inserted.

  remove a page break, right-click on the report body, then on the shortcut menu, click **Remove Page eaks** and then the corresponding page break from the sub menu.

  Notes:
  - Page breaks can be inserted into the report body level.
  - Page breaks can be inserted into banded objects only when the following conditions have been met:
    - The parent of the banded object is the report body.
    - There isn't any existing page panel in the report body.
    - Its Position property of the banded object is set to static.

- **Undoing/redoing actions**
  can undo or redo some actions by clicking **Menu** > **Edit** > **Undo** or **Redo** (or the **Undo** button or **do** button on the Standard toolbar).

- **Showing/hiding user information**
  e User Information bar shows the current user name, catalog path and name, and report path and name. can click **Menu** > **View** > **User Information Bar** to show or hide the bar.

  Tip: Administrators can make this bar open by default. To do this, log onto the JReport Administration page, click **Profile** on the system toolbar and select Customize Profile from the drop-down menu. Go to the **Page Report Studio > Properties** > **Default** tab, and then check the **User Information Bar** option.

- **Showing/hiding editing marks**
  Page Report Studio, you can use editing marks (dashed outlines of objects) for purposes such as aligning, ving and resizing. By default, the editing marks are shown only when you create a new blank report in e Report Studio. You can click **Menu** > **View** > **Editing Marks** to switch the status of the editing marks required.

- **Asking for help**
  any time, you can click **Menu** > **Help** > **User's Guide** to open the index page of Page Report Studio er's Guide. Furthermore, you can click the **Help** button in any dialog to show the help about the dialog. can also use the Help menu to open the User's Guide and access Jinfonet Software website for more ormation.

- **Exiting the report**
  ou want to close the current report and release the resources, just click **Menu** > **File** > **Exit** (or the **Exit** tton which is always on the upper right corner of the Page Report Studio window, or the close button he browser window). Closing the only report tab open will also prompt you whether or not to close the ort. In case that you have modified the report without saving it, Page Report Studio will prompt you to e the report. If you have changed the sort and/or filter criteria, you can check Sort and/or Filter in this log to save these changes with the report. Click **Yes** to save the report and close the report.

Ad hoc reporting

Ad hoc reporting allows a business analyst or end user to create a new report, add new objects to an existing report, modify report objects, and save the report or report result. These tasks are performed in the JReport Server environment and do not require use of JReport Designer.

This section describes the following ad hoc reporting tasks:

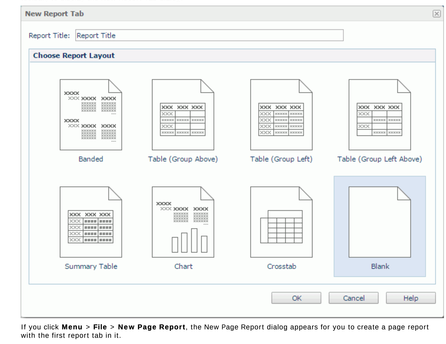
- Creating a report tab
- Adding report objects
- Applying web controls
- Making simple modifications to report objects
- Saving the report
- Exporting the report result

Creating a report tab

In Page Report Studio, you can create a new report tab based on a predefined business view to the current page report. You can also create a new page report containing one report tab and then add report tabs to it. However, the reports created on business views in Page Report Studio cannot be edited in JReport Designer any further.

To create a report tab:

1. In a Page Report Studio window, click **Menu** > **File** > **New Page Report Tab** (or the button on the Standard toolbar) to display the New Report Tab dialog.



   If you click **Menu** > **File** > **New Page Report**, the New Page Report dialog appears for you to create a page report with the first report tab in it.

2. Specify the title of the report tab as required in the Report Title text box.
3. In the Choose Report Layout box, select the required layout with which you want to create the report tab.
4. Click **OK** to create the report tab.
   - If Blank is selected as the layout, a report tab which is blank will be created. You can then use the Toolbox and the Resource View panels to add objects and view elements to the report tab.
   - If you select the layout as Banded, Table, Chart, or Crosstab, the corresponding report wizard will then be displayed. Specify the settings according to your requirements.

Also, on the JReport Console > Resources page, you can directly create a new page report (containing a report tab) in a folder into which one or more catalogs containing some business views have been published. To do this:

1. Open the folder and select the catalog for the new page report from the Catalog drop-down list.
2. On the task bar of the Resources page, click **New** > **Page Report**.
3. In the New Page Report dialog, create the page report containing a report tab as required.

The following topics show in detail how to create a report tab from a particular layout:

Creating a banded report

A banded object is a kind of component that can present grouped data and detailed data, and is composed of several banded panels with which you can easily organize data fields and other elements.

To create a banded report:

1. In the New Report Tab dialog, specify the title of the report tab in the Report Title text box, select **Banded** as the layout and click **OK** to display the Banded Wizard.
2. In the Data screen, select the business view in the current catalog, on which the banded object will be built.
3. In the Display screen, add the required fields from the Resources box to be displayed in the banded object. Modify the display name of any added field if necessary.
4. In the Group screen, add the group objects as the grouping criteria, then specify the sorting direction of each group in the Sort column.
5. To add summaries, go to the Summary screen. Select the group to which the summary will be applied, then add an aggregation object as the summary field.
6. In the Query Filter screen, specify the filter you want to apply to the business view.
7. In the Style screen, apply a style to the banded object.
8. Click **Finish** to create the report.

Creating a table report

Tables give you great control over how to present data, including placing fields, grouping them, and sorting them. A table is composed of row and columns, and each contains several cells. With such a structure a table is a good way to show any two-dimensional dataset.

To create a table report:

1. In the New Report Tab dialog, specify the title of the report tab in the Report Title text box, select the desired table type in the Choose Report Layout box, then click **OK** to display the Table Wizard.
   - **Table (Group Above)**
     Creates a table with group information above the detail panel.
   - **Table (Group Left)**
     Creates a table with group information left to the detail panel.
   - **Table (Group Left Above)**
     Creates a table with group information left above the detail panel.
   - **Summary Table**
     Creates a table with only group and summary information.
2. In the Data screen, select the business view in the current catalog, on which the table will be built.
3. In the Display screen, add the required fields from the Resources box to be displayed in the table. Modify the display name of any added field if necessary.
4. In the Group screen, add the group objects as the grouping criteria, then specify the sorting direction of each group in the Sort column.
5. To add summaries, go to the Summary screen. Select the group to which the summary will be applied, then add an aggregation object as the summary field. For the Group Left table, you can use the Row and Column columns to control the position of the summary field in the table.
6. In the Query Filter screen, specify the filter you want to apply to the business view.
7. In the Style screen, apply a style to the table.
8. Click **Finish** to create the report.

Creating a crosstab report

A crosstab summarizes data and presents the summaries in a compact row and column format.

1. In the New Report Tab dialog, specify the title of the report tab in the Report Title text box, select **Crosstab** as the layout and click **OK** to display the Crosstab Wizard.
2. In the Data screen, select the business view in the current catalog, on which the crosstab will be built.
3. In the Display screen, select a group object and click or to add it to the Columns or Rows box as a group field. Select an aggregation object and click to add it to the Summaries box as an aggregate field. Repeat this to add more group/aggregate fields.
4. In the Display Name column, edit the display names of the added group fields or aggregate fields if required. These will label the rows, columns and summaries when the report is displayed. By default these are blank and no labels will be created.
5. In the Sort column, specify the sorting manner for the group fields.
6. If you want to remove any group/aggregate field, select it and click . To adjust the order of the group/aggregate fields, select a group/aggregate field and click or .
7. In the Query Filter screen, specify the filter you want to apply to the business view.
8. In the Style screen, apply a style to the crosstab.
9. Click **Finish** to create the report.

Creating a chart report

A chart organizes and graphically presents data in a way that makes it easy for end users to see comparisons, trends, and patterns in data. It represents the report data in a visually straightforward form. A chart is based on the chart platform. On the platform, the chart paper, the legend, and labels make up the chart. You can create a chart that contains only simple DBFields, or a complicated chart that contains DBFields, groups, summaries, and even formulas. Normally, DBFields, summaries, and formulas in a report are represented in a chart using chart data markers, and groups are used to produce category names and data series names. DBFields can also be used as category names.

To create a chart report:

1. In the New Report Tab dialog, specify the title of the report tab in the Report Title text box, select **Chart** as the layout and click **OK** to display the Chart Wizard.
1. In the Data screen, select the business view in the current catalog, on which the chart will be built.
2. In the Type screen, specify the chart type as required.

   A default chart type exists in the Chart Type Groups box. To replace it with another one, select a chart type from the Chart Type box. The thumbnails of the subtypes in this type will then be displayed in the Subtype box. Select the required subtype to replace the default chart type.

   If you want to create a combo chart, click **<Add Combo Type>** of Primary Axis or Secondary Axis in the Chart Type Groups box, and an additional subtype will be added. To replace the additional subtype, select it, then specify the required type and subtype respectively in the Chart Type and Sub Type boxes.

   To add more subtypes, repeat the procedures. To remove a subtype, select it and click .
3. In the Display screen, select a group object in the Resources box and add it to the Category or Series box, the data of which will be displayed on the corresponding axis. Select a subtype in the Show Values box, then add an aggregation object or an additional value as the data of the subtype.

   To add an additional value to a subtype:

   a. Select the subtype in the Show Values box.
   b. In the Resources box, expand the **Additional Values** node, then select **Constant Value**/**Average Value**.
   c. Click beside the Show Values box. The Edit Additional Value dialog appears.
   d. In the Name text box, specify the display name for the constant/average value.
   e. Input the constant value with numeric type in the Value text box, or select a field based on which the average value will be calculated from the Based On drop-down list.
   f. Click **OK**, and the defined constant/average value will be added to the subtype.

   If you want to further modify a constant/average value, select the value in the Show Values box, then click . In the Edit Additional Value dialog, edit the value as required.

   You can add more than one aggregation object or additional value to a subtype. Each added subtype shall have at least one aggregation object or additional value.
4. If you want to define the sort order and Select N condition on the category/series axis of the chart, click the **Order/Select N** button below the Category/Series box, then define the condition in the Order/Select N dialog.

   To define a sort order and Select N condition on the category/series axis:

   a. In the Order box of the Order/Select N dialog, specify in which order values on the category/series axis will be sorted.

   b. In the Select N box, specify the Select N condition to All, Top or Bottom. If All is selected, all category/series values will be shown in the chart; if Top or Bottom is selected, the text field next to it will be enabled and you can specify an integer here, which means that the first or last N category/series values will be shown in the chart.
   c. Check the **Based On** checkbox and specify values for the two drop-down lists that follow according to your requirement.

      If Based On is unchecked, the order of the first or last N category/series values will be based on what you specify in the Order box of the dialog; if you check it, the order will be based on values of the summary field and the sort direction you specify in the drop-down lists next to the Based On checkbox.
   d. If you have selected Top or Bottom from the Select N drop-down list, you can check the **Other** checkbox and the type a character string in the next text field, so that the category/series values beyond the first or last N range will be merged into the group with the name as that character string.
   e. Click **OK** to accept the settings.
5. In the Query Filter screen, specify the filter you want to apply to the business view.
6. In the Style screen, apply a style to the chart.
7. Click **Finish** to create the report.

Notes:

- A JReport Live license for JReport Server is required in order to use this feature. If you do not have the license, contact your Jinfonet Software account manager to obtain it.
- Before you can create a report tab in Page Report Studio, you need to first make sure that the catalog corresponding to the current page report contains one or more business views and that the Pop-up Blocker is not enabled on your web browser.
- If there is only one business view in the current catalog, this business view will be used to create the report by default, and the Data screen will be hidden from the wizard. This is the same case when there is only one style available to be applied to the report.

Adding report objects

For a newly-created or an existing report, if the corresponding catalog contains business views, then you can add labels, images, banded objects, tables, crosstabs, charts, special fields and web controls to the report.

Object placement

Objects can be placed within banded objects, tables, tabulars, as well as onto an empty area of a report. The following table lists the report areas that are valid targets for the various objects, listed on the left.

	Report Layout Area								
Object	**Page Header / Footer**	**Report Header / Footer**	**Report Body**	**Banded Detail**	**Banded Page Header / Footer**	**Banded Header / Footer**	**Banded Group Header / Footer**	**Table Cell**	**Tabular Cell**
Banded object	Y	N	Y	Y	Y	Y	Y	N	Y
Chart	Y	Y	Y	Y	Y	Y	Y	N	Y
Crosstab	Y	Y	Y	Y	Y	Y	Y	N	Y
Table	Y	Y	Y	Y	Y	Y	Y	N	Y
Group object	Y	Y	Y	Y	Y	Y	Y	Y	Y
Detail object	Y	Y	Y	Y	Y	Y	Y	Y	Y
Aggregation object	N	N	Y	N	N	Y	Y	N	N
Formula	Y	Y	Y	Y	Y	Y	Y	Y	Y
Label	Y	N	Y	Y	Y	Y	Y	Y	Y
Special field	Y	N	Y	Y	Y	Y	Y	Y	Y
Image	Y	N	Y	Y	Y	Y	Y	Y	Y
Web control	Y	Y	Y	Y	Y	Y	Y	N	Y

To add an object into a report:

1. Click **Menu** > **Insert**, then click the command corresponding to the object you want to add.
2. Point to the destination where you want the object to be added, and then click the mouse button.
   - If you specify to add a label, a label will be inserted there. Edit the text of the label and format it according to your requirements.
   - If you specify to add an image, the Insert Image dialog will be displayed. Select the image you need and click **OK** to insert it into the report.
   - If you specify to add a banded object, table, crosstab, or chart, the corresponding report wizard will be displayed. Specify the settings in the wizard according to your requirements. For details, see the specific topic in Creating a report tab.
   - If you specify to add a special field, the special field will be inserted there.
   - If you specify to add a parameter control, parameter form control, or filter control, the corresponding insert control dialog will be displayed. For how to specify the settings in the dialog and the usage of the web control, see Applying web controls.
   - If you specify to add a navigation control, a navigation control will be inserted there. For the usage of the navigation control, see Applying web controls.

Alternatively, you can also use the Toolbox panel to add objects other than special fields into a report by dragging them from the panel to the destination. However, in order to use the Toolbox panel to add components, you should make sure that this ad hoc feature is enabled in the specified Page Report Studio feature profile. This setting can only be made by administrators.

Note: A JReport Live license for JReport Server is required in order to use this feature. If you do not have the license, contact your Jinfonet Software account manager to obtain it.

Applying web controls

In Page Report Studio, these four types of web controls can be applied: parameter control, parameter form control, filter control, and navigation control. This section describes each of the web controls and how to use them.

Note: A JReport Live license for JReport Server is required in order to use this feature. If you do not have the license, contact your Jinfonet Software account manager to obtain it.

Using parameter control to specify a parameter to a report

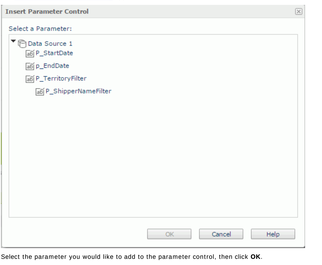
A parameter control is a web control that is bound with a parameter used by the current report. By specifying values to the parameter in a parameter control, you can pass the parameter values to JReport and run the report with the specified values.

Cascading parameters cannot be used in parameter controls. If you want to do this, use parameter form controls instead.

To insert a parameter control and use it to specify a parameter to a report:

1. Do either of the following:
   - Click **Menu** > **Insert** > **Parameter Control**, then point to the destination where you want to add the parameter control and click the mouse button.
   - Drag **Parameter Control** from the Toolbox panel to the destination in the report.

   The Insert Parameter Control dialog appears.



2. Select the parameter you would like to add to the parameter control, then click **OK**.
3. A parameter control will be added into the report. Specify the value for the parameter. You may specify the value in one of these ways.
4. Once the value in the parameter control changes, the report will rerun with the new parameter value.

Note: If the specified parameter is no longer used in the report, the parameter control will become invalid.

Using parameter form control to run reports

A parameter form control is a web control that is bound with the parameters used by the current report or other reports. By specifying values to the parameters in a parameter form control, you can make the reports run with the specified parameter values.

To insert a parameter form control and use it to run report:

1. Do either of the following:
   - Click **Menu** > **Insert** > **Parameter Form Control**, then point to the destination where you want to add the parameter form control and click the mouse button.
   - Drag **Parameter Form Control** from the Toolbox panel to the destination in the report.

   The Insert Parameter Form Control dialog appears.

2. Specify the target reports to run using the parameter form control.
   - To run the current report, select **Current Report**, then specify the parameters used to run the report from the Select Parameters box.
   - To run other reports, select **Others**, then select the reports you want to run. If all the selected reports contain no parameters, you cannot finish the dialog.
3. Specify whether to include the Submit button in the parameter form control. If Submit is included, it is used to submit the parameter values you specified in the parameter form control. If Submit is not included, once you change the values of a parameter in the parameter form control, the new values will be applied automatically.
4. Click **OK** in the dialog to save the changes.

   The parameter form control is now inserted in the report. It lists the selected parameters for the current report or lists all parameters used by the specified reports.
5. In the parameter form control, specify values of the listed parameters. You may specify the values in these ways.
6. Click the **Submit** button to run the current report or the specified reports if the button is available. If there is no Submit button, the change of values in the parameter form control will trigger report rerunning.

Note: If you save or publish a report containing a parameter form control to another directory, the reports that you bind the parameter form control with will not be saved or published along with the report.

Using filter control to filter report data

A filter control is used to filter one or more data components in a report, which refer to tables, banded objects, charts, and crosstabs. For how a filter control works, see Filtering scenarios.

To insert a filter control and use it to filter report data:

1. Do either of the following:
   - Click **Menu** > **Insert** > **Filter Control**, then point to the destination where you want to add the filter control and click the mouse button.
   - Drag **Filter Control** from the Toolbox panel to the destination in the report.

   The Insert Filter Control dialog appears.

2. From the resource list, select the fields of the same data type to bind to the filter control.

   To filter components created from the same data source, select a field in the data source.

   To filter components created from different data sources, find a common field these data sources contain, then select the field in each of the data sources.
3. The Apply To drop-down list provides the components involving the selected fields. Select the components which you want to filter.
4. When done, click **OK**.

   The filter control is inserted in the report. It lists all values of the specified fields. You can select one or more values to apply.

After inserting filter controls in the report, you can also insert a navigation control for undoing/redoing the value selection in the filter controls. For details about the usage of navigation control, see Using navigation control to undo/redo value selection in filter controls.

Managing a filter control

After right-clicking on the title bar of a filter control, these options are available for managing the filter control.

- **Properties**
  Opens the Filter Control Properties dialog for editing the properties of the filter control. The dialog provides an entry to changing the data fields that the filter control is based on and the application scope. To do this, in the General tab, click [...] beside the Filter On text box. Then in the Select Field dialog, change the settings as you like and click **OK**.
- **Search**
  Displays the quick search toolbar which enables you to search values in the filter control. You can also click the button on the title bar of the filter control to launch the quick search toolbar.

The following are details about the usage of the quick search toolbar:

- **Text field**
  Type in the text you want to search for in the text field and the matched text will be highlighted among the field values.
- **X**
  Closes the quick search toolbar. You can also click outside of the quick search toolbar to achieve this.
- **+**
  Lists the advanced options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that matches a whole word and ignore partial word matches.
- **Highlight the next matched text.**
- **Highlight the previous matched text.**
- **Clear**
  Cancels the selection of values in the filter control. You can also use the button on the title bar to cancel the selection. This operation can be undone/redone.
- **Sort**
  Sorts the values in the filter control in the ascending or descending order.
- **Delete**
  Removes the filter control from the report and the filter you created with the filter control will be removed from the report too. You can also use the close button on the title bar to remove the filter control.
- **Hide**
  Hides the filter control.

Cascading relationship between filter controls

When there are filter controls that apply to the same data components, and when these controls' fields have cascading relationship, the cascading relationship will be revealed when you select values in the controls.

For example, there is a filter control based on the field Country, a filter control on City, and another on State. The first two share one table while the third shares nothing with the other two. In this case, Country and City values will show cascading relationship, but State values will not participate. You select USA in the Country filter control, the values in the City filter control will change as follows if the control has scrollbar: the cities belong to USA are displayed in the upper area of the filter control, and the other cities are put in the lower area and grayed out. For the case that the City filter control has no scrollbar: all the values remain their positions and the values not belonging to USA are grayed out. In both cases all the values are selectable. But the State values remain as before, since the selection of them will not affect the data components that the Country and City filter controls control.

Using navigation control to undo/redo value selection in filter controls

A navigation control can be considered as an accessorial control for filter controls and is used to deal with the value selection operations in all the filter controls in the same report.

To insert a navigation control into a report, do either of the following:

- Click **Menu** > **Insert** > **Navigation Control**, then point to the destination where you want to add the navigation control and click the mouse button.
- Drag **Navigation Control** from the Toolbox panel to the destination in the report.

A navigation control is a combination of three buttons:

- **Back**
  Goes back to the previous value selection status and refreshes the report data accordingly.
- **Clear**
  Removes all the value selection histories and all the filter conditions based on the selections, and refreshes the report data accordingly.
- **Forward**
  Goes forward to the next value selection status and refreshes the report data accordingly.

Making simple modifications to report objects

By virtue of Page Report Studio's powerful ad hoc functions, you can make simple modifications to report objects at runtime while viewing the report in Page Report Studio.

Note: Except for showing/hiding objects, a JReport Live license for JReport Server is required in order to use all the other features introduced in this document. If you do not have the license, contact your Jinfonet Software account manager to obtain it.

Moving an object

A table, banded object, chart, crosstab, tabular, or image, can be easily moved to a new position. What you need to do is click anywhere in the object, then drag the icon appearing at its upper left corner to the destination. After Page Report Studio has finished processing, the object will be redrawn in the new location.

For other objects, select it and move it to the new position.

Notes:

- Before you can move any object in a report, you need to first make sure that the Page Report Studio window is in the Interactive View mode.
- For reports designed in JReport Designer, only the objects whose Position property value is absolute, and the DBFields or labels which have been defined as a view element can be moved in Page Report Studio.

Resizing an object

To resize an object, click anywhere in the object, when the icon appears at its upper left corner, click the icon to select the object, then you will see that it is surrounded by a rectangle with three resizing handles. Point to a handle, when the mouse pointer turns to a double-headed arrow, you can drag the handle to resize the object.

To resize a panel in a banded object, select it and drag the resizing handle to the desired position.

To adjust the width/height of a column/row in a table, point to the right/lower boundary of the column/row, when the mouse pointer becomes a horizontal/vertical double-headed arrow, drag the handle and the width/height of the column/row will change. This will also resize all cells in the column or row.

For a crosstab, you can resize its rows and columns the same as you do with a table.

For a tabular, point to the boundary between two cells and the mouse pointer will become a double-headed arrow, you can then drag the boundary to adjust the size of the related cells.

To change the width and height of a field, click any value of this field to select it, then drag the right or lower resizing handle on its borders to a new position, and the width or height of the field will change. You can also do this for any label.

Notes:

- When resizing table rows:
  - If you resize the table header, only the height of the header will be changed. However, when you resize any row except the header, the height of all rows in the table will be changed at the same time.
  - If there are some groups in a table and the height of one group row is changed, the other group rows will not be resized.
- When resizing crosstab columns/rows:
  - If you resize the horizontal/vertical header of a crosstab, other rows/columns will not be affected.
  - If you resize the total column/row of a crosstab, other columns/rows will not be affected.

Hiding/showing an object

To hide a table, banded object, chart, crosstab, or tabular, click on the object, when the icon appears at its upper left corner, right-click on the icon and then select **Hide** from the shortcut menu. For other objects such as text boxes, drop-down lists, fields, and labels, right-click it and then select **Hide** to hide it.

To show a hidden object, right-click the object containing it, then on the shortcut menu, select the object name from the Show submenu.

Splitting and merging cells in a tabular

Adjacent cells in a tabular which can form a rectangle may be merged into one cell.

To merge adjacent cells, select them one by one while holding the Ctrl key, then click **Menu** > **Report** > **Merge**, and these cells will be merged into one cell.

To split a cell:

1. Select the cell and click **Menu** > **Report** > **Split**.
2. In the Split dialog, specify the number of rows and columns.
3. Click **OK** and the cell will be split.

Modifying object properties

Page Report Studio allows you to modify object properties with the corresponding properties dialog. However for some objects, you must be an advanced user in order to edit their properties.

- To edit the properties of any object in a report, right-click on the object and select **Properties** from the shortcut menu. If it is a table, crosstab, chart, banded object, or tabular, click anywhere on it, when the icon appears at its upper left corner, right-click on the icon and click **Properties** on the shortcut menu. In the corresponding properties dialog, specify the settings as required. For a table, you can also right-click any field or cell in it and select **Table** from the shortcut menu to show the Table Properties dialog. For detailed explanation about options in the properties dialogs, refer to the specific topics in Page Report Studio dialogs.
- You can right-click a group header/footer panel in a banded object, and then select **Group** to show the Group Properties dialog in order to define the group properties.
- To edit the properties of a report tab, right-click the blank part of the report tab and select **Report** from the shortcut menu, then in the Report Tab Properties dialog, configure the properties as required.

Tip: If you just want to modify the text related properties for a field or label, for example, you want to change the text alignment or make the text bold, you can achieve it by simply selecting the field or label, then clicking the corresponding buttons on the toolbar.

Deleting an object

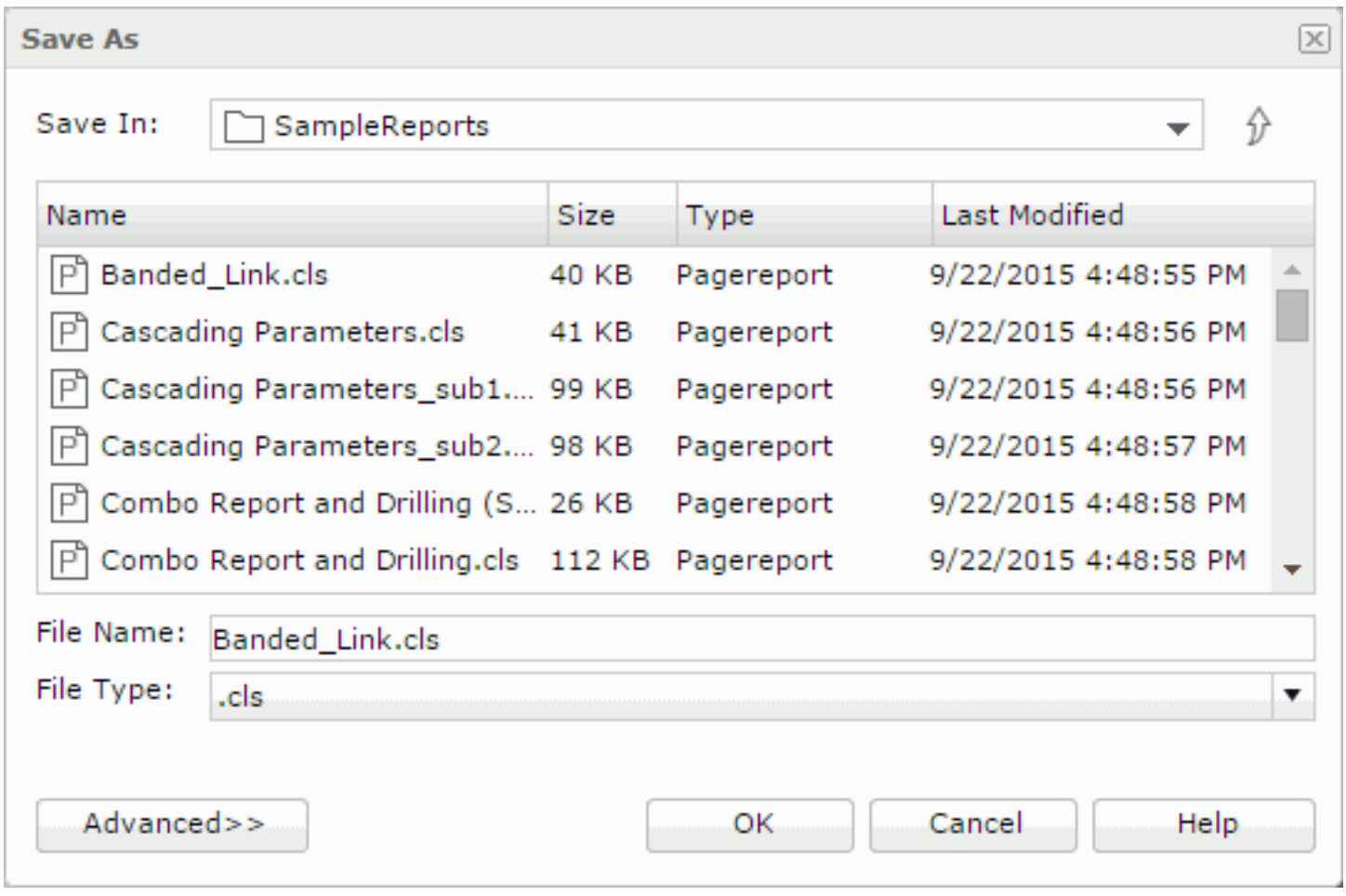
An object can be removed from the report if it is no longer required. However, objects that are in a subreport cannot be deleted.

- To delete a table, banded object, chart, crosstab, tabular, or image, click on the object, when the icon appears at its upper left corner, right-click on the icon and select **Delete** from the shortcut menu, or you can drag the icon outside the report page. Then, a message box will prompt, asking for your confirmation. Click **OK** in the message box so as to remove the component.
- For a field, you can drag any value of the field outside the report page to remove it. You can also drag any label outside the report page to remove it. Right-clicking and then selecting Delete is another way to achieve this.

Saving the report

You can save your report in a Page Report Studio window. To do this, click **Menu** > **File** > **Save** (or the **Save** button on the Standard toolbar). The Save Report Template dialog appears. The Sort and Filter options in this dialog signify whether or not to include the sort and filter criteria when saving. Specify the options as required, then click **OK**, and the report will be saved as a report version.

If the report is newly created and has not yet been saved, the Save As dialog appears.



1. In the Save in section, browse to the folder where you want to save the report in the server resource tree. The folder may be Public Reports or My Reports. You can use the button to go to the parent folder.

   The resource table shows the resources in the current directory. Click the column names to change the order of the report in the table list if required.

2. In the File Name box, enter the name of the report or use the default name.

3. From the File Type drop-down list, specify the type of the saved report.

4. Click the **Advanced** button to set the advanced settings for the report if required.

   a. The catalog that the report uses is shown.

      Specify the relationship between the saved report and the catalog used to run it (activated only when Select Catalog Linked Model is checked in the Profile > Customize Profile > Page Report Studio > Properties > Advanced tab):

      - **Set Original Catalog as Linked Catalog into Saved Page Report**
        If checked, the saved report will be linked with the catalog and the saved report will run with the catalog no matter whether the two are in the same directory. If later the catalog is updated, the saved report will run with the latest version of the catalog.

      - **Set Catalog Copy to Public Reports / My Reports**
        If checked, the catalog will be copied to the directory where the report is saved and the saved report will run with the copied catalog.

   b. If you want to save the report together with the sort and filter criteria, check **Save Sort Criteria** and **Save Filter Criteria** correspondingly. With the criteria saved, Page Report Studio will automatically apply them to the report the next time it is opened.

   c. Optionally, input comments in the Description box as a description for the report.

5. Click **OK** to save the report.

To save a copy of a report, click **Menu** > **File** > **Save As** (or the **Save As** button on the Standard toolbar) to show the Save As dialog, and then do as above.

Notes:

- You will not be able to save the report to some locations if you do not have the required permissions. You need to have Write access to the directory.
- If one of the report tabs in a report contains subreports, when you save the report, changes you have made on the subreports will not be saved along with the primary report.
- To find a newly saved report version, browse to select the row that the report is in on the JReport Console > Resources page, click **Tools** > **Version** on the task bar, and then click the **Report Versions** tab.

Exporting the report result

When you are satisfied with the result of the active report, you may want to export it as a result version or as a local file to other formats.

The following topics describe exporting report result in detail:

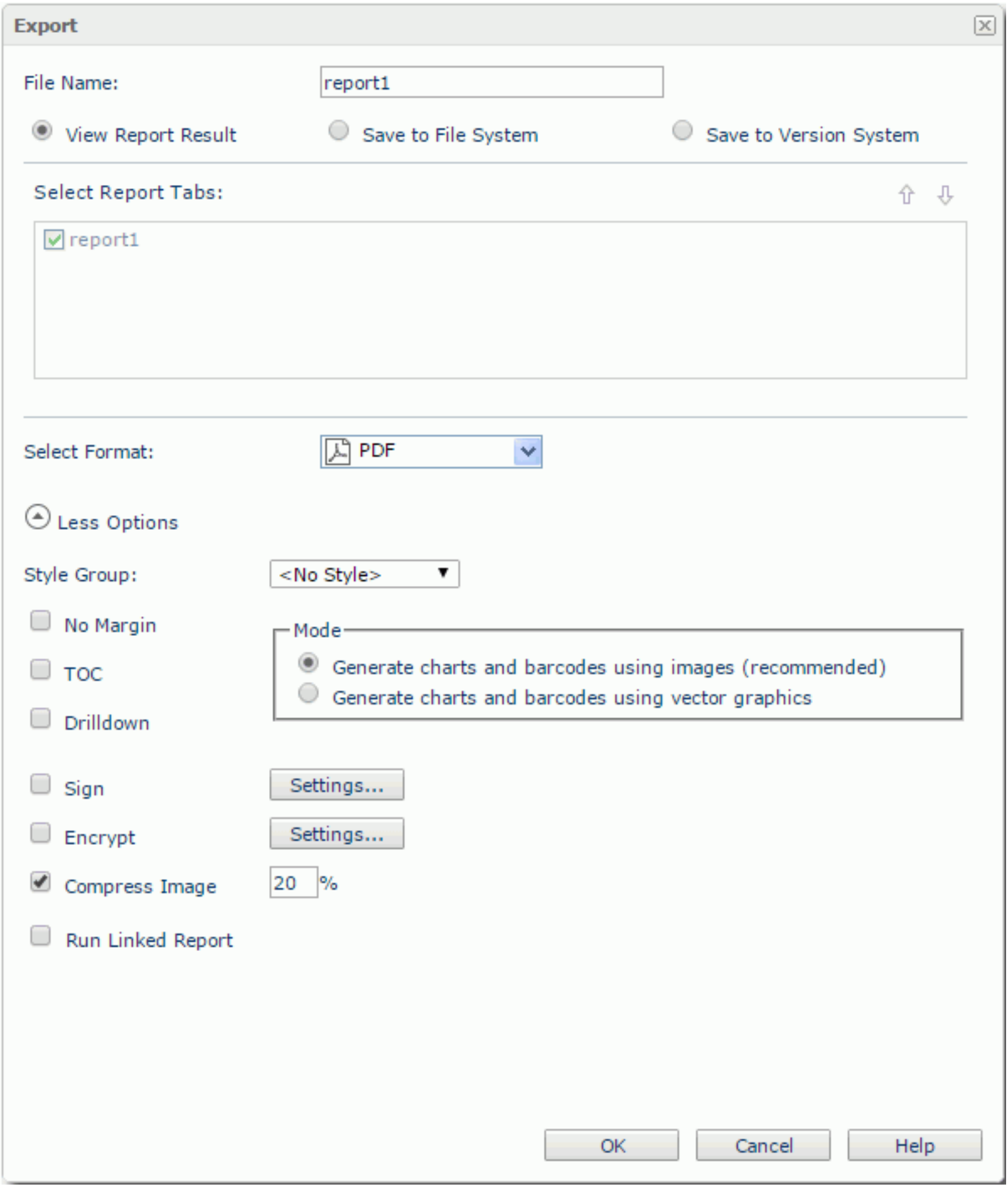
- Exporting the result from a Page Report Studio window
- Exporting the result by using a JavaScript function
- Customizing buttons for one-step exporting
- Controlling user access to different export formats
- Customizing warning messages

Note: If the report you are going to export is linked to another report, in the exported results, the link will no longer be available.

Exporting the result from a Page Report Studio window

You can export the report result from a Page Report Studio window to other formats by taking the following steps:

1. Click **Menu** > **File** > **Export** (or the **Export** button on the Standard toolbar) to display the Export dialog.



2. In the File Name field, specify the name of the exported result file.
3. Specify the destination of the result:
   - **Save to Version System**: The result will be saved as a result version in JReport Server's versioning system.
   - **Save to File System**: The web browser will prompt you to save the result file to a specified folder. If selected, you need to provide a name for the result file in the File Name field.
   - **View Report Result**: The result will be directly opened in the web browser if the format is supported by a plug-in of the web browser; otherwise it will prompt you to save the result file.
4. In the Select Report Tabs box, specify the report tab(s) which you want to export. The selected report tabs will be exported in the list order. You can change the report tabs order by clicking or .
5. From the Select Format drop-down list, select the format in which to export the result: HTML, PDF, Excel, Text, RTF, XML, PostScript, or Page Report Result.
6. To specify the additional setting of the selected format, click **More Options**.
7. From the Style Group drop-down list, select the style group you want to apply to the exported report result. If No Style is selected, the style group property predefined for the specified export format in JReport Designer will be applied to export the report result to that format.
8. Set the other properties for the selected format as required (for details about properties of each format, see Export dialog).
9. Click **OK** to confirm.

Exporting the result by using a JavaScript function

JReport Server provides you with a JavaScript function which allows you to open the report result or export it in a specified format. You can find this function in the file API.js in `<server_install_root>\public_html\javascript\dhtml`, as shown below:

```
function user_oneStepExport(type, options)
```

The following explains this function's two arguments, type and options, in detail.

type - Specifies the export format.

- HTML = 0
- PDF = 2
- PS = 3
- RTF = 4
- TEXT = 5
- EXCEL = 6
- XML = 7

options - Specifies the values of the options of each format. It is a string array whose member is of the format "key=value". The options and their usage are listed as follows:

Key	Description	Available Value	Default Value
HTML			
to_ver	Specifies whether or not to save the result to version.	true, false	false
to_open	Specifies whether or not to export and open the result file.	true, false	false
to_local	Specifies whether or not to save the result to a local file.	true, false	false
browser	Specifies the web browser type.	0 - IE or Chrome 1 - Firefox	0
imagetype	Specifies the type of the images in the result file.	0 - Decided by JReport 1 - GIF 2 - JPG	0
overflow	Specifies the overflow type.	0 - VISIBLE 1 - HIDDEN 2 - VERFLOWCOUNT	0
resolution	Specifies the HTML resolution.	Any integer between 1 and 4294967296	96
title	Specifies the title for the HTML file.	Any string	""
applet	Specifies whether or not to export chart in Java applet format.	true, false	true
css	Specifies whether or not to embed the cascading style sheet in the exported HTML files.	true, false	false
multi	Specifies whether or not to generate an HTML file for each page of the report result.	true, false	false
hyperlink	Specifies whether or not to contain hyperlinks in the HTML file.	true, false	false
pagenumber	Specifies whether or not to contain page numbers in the HTML file.	true, false	false
drilldown	Specifies whether or not to include the drilled-down file in the exported HTML file.	true, false	false
no_margin	Specifies whether or not to remove the original margins.	true, false	false
absolute	Specifies whether or not to make the font size fixed in the web browser.	true, false	false
PDF			
to_ver	Specifies whether or not to save the result to version.	true, false	false
to_open	Specifies whether or not to export and open the result file.	true, false	false
to_local	Specifies whether or not to save the result to a local file.	true, false	false
no_margin	Specifies whether or not to remove the margin.	true, false	false
simulate	Specifies whether or not the mode is to be Simulated Printing Mode.	true, false	true
standard	Specifies whether or not to set the mode as Standard Mode.	true, false	false
content	Specifies whether or not to contain the TOC in the exported PDF file.	true, false	false
drilldown	Specifies whether or not to include the drilled-down file in the exported PDF file.	true, false	false
encrypt	Specifies whether or not to encrypt the report result.	true, false	false
compress	Specifies whether or not to compress the images in the report.	true, false	false
ratio	Specifies the percentage with which to compress the images in the report.	Any integer between 1 to 100	20
compatibility	Specifies the encryption compatibility.	0 - Acrobat 3.0 and later 1 - Acrobat 5.0 and later	1
doc_psw	Specifies the password for opening the PDF file when encrypt=true.	Any string	""
permi_pasw	Specifies the password for printing and editing the PDF file when encrypt=true.	Any string	""
printing	Specifies the PDF printing mode.	0 - Prevents users from printing the file 4 - Allows low resolution-printing 2052 - Allows high-resolution printing	0
changes	Defines which editing actions are allowed in the PDF file.	0 - Prevents users from making any changes to the file 1024 - Allows inserting, deleting, and rotating pages 256 - Allows users to fill in form fields and adding digital signatures. 32 - Allows users to fill in form fields and add digital signatures and comments 40 - Allows users to do anything except extracting pages 2108 - Allows all	0
enable_copy	Specifies whether or not to allow users to copy the file contents.	true, false	false
enable_access	Specifies whether or not to let visually impaired users read the document with window readers.	true, false	true
PostScript			
to_ver	Specifies whether or not to save the result to version.	true, false	false
to_open	Specifies whether or not to export and open the result file.	true, false	false
to_local	Specifies whether or not to save the result to a local file.	true, false	false
no_margin	Specifies whether or not to remove the margins in the PS file.	true, false	false
RTF			
to_version	Specifies whether or not to save the result to version.	true, false	false
to_open	Specifies whether or not to export and open the result file.	true, false	false
to_local	Specifies whether or not to save the result to a local file.	true, false	false
rtf_flow	Specifies whether or not to apply a flow layout when exporting the report to RTF.	true, false	false
no_margin	Specifies whether or not to remove the margins in the RTF file.	true, false	false
Text			
to_ver	Specifies whether or not to save the result to version.	true, false	false
to_open	Specifies whether or not to export and open the result file.	true, false	false
to_local	Specifies whether or not to save the result to a local file.	true, false	false
repeat	Specifies whether or not to replace a field value of a record with that of its previous record if the field value is null.	true, false	false
compress	Specifies whether or not to compress the clearance between columns.	true, false	false
win_linebreak	Specifies whether or not to use Windows end-of-line characters.	true, false	true
normal	Specifies whether or not to generate the report result to a standard text file.	true, false	true
quote_mark	Specifies whether or not to mark the fields in the exported file with quotation marks.	true, false	false
head_foot	Specifies whether or not to contain all headers and footers in the report.	true, false	true
delimiter	Specifies the delimiter.	Any single character	
width	Specifies the user-defined character width.	An integer	
height	Specifies the user-defined character height.	An integer	
Excel			
to_ver	Specifies whether or not to save the result to version.	true, false	false
to_open	Specifies whether or not to export and open the result file.	true, false	false
to_local	Specifies whether or not to save the result to a local file.	true, false	false
wrap	Specifies the word-wrap setting.	0 - All Keep Existing 1 - All Disabled 2 - All Enabled	0
new_layout	Specifies whether or not to use the new layout mode.	true, false	true
shapes	Specifies whether or not to include the shapes in the exported file.	true, false	false
excel_2000	Specifies whether or not to export the result in Data Format, which means only the report data will be exported without format.	true, false	false
advanced	Specifies whether or not to apply the advanced options.	true, false	false
header	Specifies the page header text.	Any string	
footer	Specifies the page footer text.	Any string	
gridline	Specifies whether or not to print gridlines when printing the exported Excel file.	true, false	false
XML			
to_ver	Specifies whether or not to save the result to version.	true, false	false
to_open	Specifies whether or not to export and open the result file.	true, false	false
to_local	Specifies whether or not to save the result to a local file.	true, false	false
only_data	Specifies whether or not to only contain the database column information in the exported XML file.	true, false	false
schema	Specifies the name of an existing schema file with its full path with which to generate the XML file.	An existing schema file with its full path.	

Notes:

- The three options to_ver, to_open, and to_local are mutually exclusive, that is, only one of them should be true, and the rest false. If you set two or three of them to true, JReport will only accept the first true and treat the rest as false. If you set all three to false, JReport will consider to_open as true.
- In the string array options, the members are separated by commas (","). You can arrange the members in any order, and omit those keys where you want to use the default value. For example, you can define a variable html_options:

  ```
  var html_options = ["imagetype=1", "resolution=120", "hyperlink=true",
  "no_margin=false", "title=hello world"];
  ```

  Then, you can use `user_oneStepExport(0, html_options)` to obtain a URL for exporting the report result in HTML format.
- JReport also provides the function user_downloadReport(type, options) whose usage is similar to user_oneStepExport.

Customizing buttons for one-step exporting

Seven buttons are provided for one-step exporting. They are Export to PDF, Export to Excel, Export to RTF, Export to HTML, Export to Text, Export to PS, and Export to XML. By default, the buttons are invisible on the toolbar area. You have to add them to the toolbar area by modifying the index.jsp file in `<install_root>\public_html\dhtmljsp`.

Use the following two API functions to control the buttons:

- public void customizeToolbar(String SessionId, String RptSetId, String toolbarname, int[] buttonId);
- public void customizeToolbar(String SessionId, String RptSetId, String toolbarname, int[] buttonId, boolean isVisible);

Explanation of these parameters:

- **SessionId** - session ID, which can be retrieved by DHTMLUtil.getSessionID(request).
- **RptSetId** - report ID, which can be retrieved by obDHTMLUtil.getRptSetId(request).
- **toolbarname** - toolbar name, such as "Standard", "Analysis", "View", or user-defined ones.
- **buttonId[]** - toolbar button ID. It is an integer array, so you need define an integer array variant.

  For example: int[] mybuttonid ={DHTMLConstant.TOOLBAR_EXPORTTOPDF, DHTMLConstant.TOOLBAR_EXPORTTOXLS, TMLConstant.BTN_EXPORT_TO_RTF}

  The array elements available for one-step exporting are:

  - DHTMLConstant.TOOLBAR_EXPORTTOPDF
  - DHTMLConstant.TOOLBAR_EXPORTTOXLS
  - DHTMLConstant.TOOLBAR_EXPORTTORTF
  - DHTMLConstant.TOOLBAR_EXPORTTOHTML
  - DHTMLConstant.TOOLBAR_EXPORTTOTEXT
  - DHTMLConstant.TOOLBAR_EXPORTTOPS
  - DHTMLConstant.TOOLBAR_EXPORTTOXML
- **isVisible** - whether or not to show the buttons defined with buttonID[]. true -- show, false -- hide. This parameter can be absent, and then the value is true.

For example, if you want to add the Export to HTML button to the Export toolbar, then:

1. Open the file index.jsp.
2. Add the code

   ```
   int[] temparray = {DHTMLConstant.TOOLBAR_EXPORTTOHTML};
   dhtmlConfig.customizeToolbar(SessionID, RptSetId, "Standard", temparray, true);
   ```

   before

   ```
   //<!-- Tool Bar -->
   if(dhtmlConfig.isFeatureEnabled(SessionID, RptSetId, DHTMLConstant.
   FEATURE_TOOLBAR)){
   ```

3. Start JReport Server.
4. Run a report in Page Report Studio, and you will see a new button Export to HTML is displayed on the Export toolbar.

To add more than one button to the toolbar area, for example, adding Export to HTML and Export to PDF buttons to the Export toolbar:

1. Open the file index.jsp.
2. Add the code

   ```
   int[] temparray = {DHTMLConstant.TOOLBAR_EXPORTTOHTML, DHTMLConstant.
   TOOLBAR_EXPORTTOPDF};
   dhtmlConfig.customizeToolbar(SessionID, RptSetId, "Standard", temparray, true);
   ```

   before

   ```
   //<!-- Tool Bar -->
   if(dhtmlConfig.isFeatureEnabled(SessionID, RptSetId, DHTMLConstant.
   FEATURE_TOOLBAR)){
   ```

3. Start JReport Server.
4. Run a report in Page Report Studio, and you will see the buttons Export to HTML and Export to PDF are displayed on the Export toolbar.
5. Click the added button on the toolbar of the report page, the report will be exported to the corresponding format, using the default values of the format options.

Controlling user access to different export formats

JReport introduces an access control ability which can restrict different users to export report result to different formats.

How to realize the access control ability

You can restrict the report result formats for users by passing values to a variable. That is the global variable enableTypes in both customize_panel.jsp and save_result.jsp files in `<install_root>\public_html\dhtmljsp` folder. The value of enableTypes should be Integer, and JReport gives you the following integers to represent the corresponding report result types:

Report Result Format	Int. Value
HTML	0
PDF	1
Excel	2
Text	3
RTF	4
XML	5
PostScript	6

You can specify a value or an array to enableTypes using the integers in the above table. Or you can also use the integers which are not mentioned above, but if you do so, the Int. values will be ignored, and do not function. That is to say, if you pass the array {1,2,5,9,-2} to enableTypes, 9 and -2 will be ignored and {1,2,5} will work, so PDF, EXCEL, XML formats can be in use.

Example

Assume that there are two users, user1 and user2. When user1 logs on, run a report in Page Report Studio, he can only export the report result as PDF and HTML formats. While user2 can export the report result as RTF and XML formats when logging on. To realize it, you can use the following codes in customize_panel.jsp or save_result.jsp file:

```
int[] enableTypes=null;
String UserName=null;
UserName=DHTMLUtil.getUserName(request);
if(UserName.equalsIgnoreCase("user1"))
enableTypes=new int[]{1,0};
if(UserName.equalsIgnoreCase("user2"))
enableTypes=new int[]{4,5};
```

Customizing warning messages

You can customize the warning messages when exporting the report result to XML through Page Report Studio. With this function, you can specify what you want to show as a warning message. The custom warning messages are supported across web browsers.

To customize the warning messages, you need to customize the jsp files using either of the following two methods:

Method 1

1. Open the save_result.jsp file in the `<install_root>\public_html\dhtmljsp` folder with your favorite editor.
2. Specify the value of customMsgForXML in the save_result.jsp to whatever you want. The value cannot be "" or null.
3. Run a report in Page Report Studio, click **Menu** > **File > Export** to display the Export dialog.
4. Select **XML** from the Select Report Result Format drop-down list, and if necessary, modify other properties as required, then click **OK**.
5. A warning message shows as you have defined.

Method 2

You can also export the report result via the Export panel instead of clicking Menu > File > Export. By default, the panel is hidden. So, the following is another way to create custom warning messages:

1. Open the customize_panel.jsp file in the `<install_root>\public_html\dhtmljsp` folder with your favorite editor.
2. Set the class of panelDIV to visibleMargin to show the Export panel on web browser.
3. Specify the value of customMsgForXML in customize_panel.jsp.
4. Run a report in Page Report Studio, and the Export panel together with the report shows.
5. Select **XML** from the Select Report Result Format drop-down list, and if necessary, modify other properties as required, then click **OK**.
6. A warning message shows as you have defined.

For example, using method2, when you need to export the report result as XML, and if you want to show "This is IE Browser" while browser is IE, you need to set panelDIV in customize_panel.jsp file to visibleMargin as follows:

```
<div id="<%=DHTMLConstant.DHTML_PREFIX%>panelDIV" class="visibleMargin">
```

And then customize CustomMsgForXML in the same jsp file as follows:

```
String browserName=request.getHeader("User-Agent");
String customMsgForXML ="This is "+browserName;
```

Access JReport Server via IE, run a report in Page Report Studio, select **XML** from the Select Report Result Format drop-down list in the Export panel, and click **OK**. Then a pop-up box will show you "This is IE Browser".

Analytic reporting

Page Report Studio provides you with a convenient and powerful tool to analyze your business information. By providing secure web access to business data and making the data interactive, Page Report Studiofacilitates data analysis.

Page Report Studio enhances the utility of production reports by making them interactive - allowing you to define your view of data to make it more useful. Through a user-friendly web GUI, report contents can be easily navigated, drilled, and viewed in detail.

Page Report Studio uses the Resource View panel to provide a business-oriented view of databases. This view shields end users from having to understand database connectivity and SQL syntax while allowing IT professionals to maintain control of business data and to ensure its integrity. Using the Resource View panel, Page Report Studio dynamically builds SQL statements to retrieve data and automatically generate multidimensional data views. These views contain the underlying data structure which makes data analysis possible.

The following topics describe the analytic reporting features:

- An introduction to business views
- Applying filters
- Using view elements
- Using dynamic resources
- Drilling through the report data
- Manipulating data components
- Adding conditional formats to fields
- Converting between components
- Sorting report data
- Searching for text in a report

Note: A component created in JReport Designer is based on a dataset, while that created in Page Report Studio is based on a business view. For the former, if you want to do analytic actions in Page Report Studio, such as adding a view element, converting the component type, drilling it, or changing chart definition, Page Report Studio will need to convert its fields to view elements.

An introduction to business views

A business view, which is needed for creating multidimensional data views, contains database connections and relationships between view elements. The business view shields report end users from having to understand the physical structure of a data source, and enables them to build reports and analyze data based on a set of view elements they can understand. It also enables IT professionals to maintain control of the business data and ensure its integrity, while presenting end users with an intuitive view of the underlying data structures.

To make use of a business view, you need to first define it at report design time in JReport Designer.

A business view may contain category objects and view elements (group objects, aggregation objects, and detail objects). You can insert these view elements or remove them to change the report result when you view reports in Page Report Studio.

- **Category objects**
  Category objects contain a collection of view elements. A business view may contain more than one category. In the Resource View panel, the icon indicates that an object is a category. Categories are only for categorizing view elements, and they cannot be inserted into a report. The category is often used for indicating the name of the underlying DBMS table.

- **Group objects**
  Group objects are view elements that will become the basis for analysis in a report. They characteristically return text or date values. In the Resource View panel, the icon indicates that an object is a group object. A group object can be inserted wherever a group field can be inserted into. It can be inserted as a column or row field in a crosstab, or as a group field or detail field in a banded object or a table, or displayed as category/series field in a chart.

- **Aggregation objects**
  Aggregation objects are numeric view elements that are calculated dynamically at runtime. The icon indicates that an object is an aggregation object. An aggregation object can be inserted wherever a summary can be inserted. For instance, it can be inserted into the group header or footer panel in a table or banded object, or into a crosstab as an aggregate field. An aggregation object can also be used as a detail field in a banded object or table although it will display the same aggregate value for every detail line. Page Report Studio will calculate the summary values based on the group level the aggregation object has been inserted into.

- **Detail objects**

  Detail objects provide additional information. The icon indicates that an object is a detail object. It can be inserted wherever a DBField can be inserted. For example, you can insert a detail object into a table or banded object as a detail field.

Applying filters

You can apply some filter to a page report so as to narrow down the data displayed in the report. The filters can be applied to the business views used by data components such as banded objects, tables, crosstabs and charts in the report, or to the data components themselves.

Applying filters to business views

For data components created on business views in a page report, you can apply some filters to the business views they use to narrow down the data scope of the business views. Filters for business views are defined into two categories in Page Report Studio: predefined filters and user defined filters. As the name suggests, predefined filters are defined on business views in advance in JReport Designer, and user defined filters are created on business views while they are used in Page Report Studio.

To apply a filter to the business view a data component uses:

1. Select the component by clicking anywhere in it, and then clicking the icon ⊞ at the upper left corner of the component.
2. Click **Menu** > **Report** > **Query Filter**, or right-click the component and select **Query Filter** from the shortcut menu to display the Query Filter dialog.

   All the predefined filters of the business view the data component uses are listed in the Query Filter drop-down list. Choose the one you want to apply. If you want to further edit the filter, click the **Edit** button and then redefine the filter as required. The edited filter will then be saved as a user defined filter to the business view.

   If you prefer to define a filter on your own, select **User Defined** from the Query Filter drop-down list, then define the filter according to your requirements.

   There are the basic and advanced modes of the dialog for you to define either simple or complex filter expressions.

   - **To define a filter using simple expressions:**
     a. Make sure the dialog is in the basic mode.

     b. From the field drop-down list, select the field on which the filter will be based.
     c. From the operator drop-down list, set the operator with which to compose the filter expression.
     d. Type the values of how to filter the field in the value text box, or select one or more values from the drop-down list.
     e. If you want to add another condition line, from the logic operator drop-down list,
     - To add a condition line of the AND relationship with the current line, select **AND**, then define the expression as required.
     - To add a condition line of the OR relationship with the current line, select **OR**, then define the expression as required.

     Repeat this to add more filter expressions if required. To delete a condition line, click ⊗ on its left.
   - **To define a filter using complex expressions:**
     a. Switch the dialog to the advanced mode.

     b. Click the **Add Condition** button to add a condition line.
     c. From the field drop-down list, select the field on which the filter will be based.
     d. From the operator drop-down list, set the operator with which to compose the filter expression.
     e. Type the values of how to filter the field in the value text box, or select one or more values from the drop-down list.
     f. To add another condition line, click the **Add Condition** button and define the expression as required. Then, click the logic button until you get the required logic to specify the relationship between the two filter expressions. The logic can be AND, OR, AND NOT, or OR NOT.
     g. Repeat the above steps to add more filter expressions if necessary.

     To group some conditions, select them and click the **Group** button, then the selected conditions will be added in one group and work as one line of filter expression. Conditions and groups together can be further grouped. To take any condition or group in a group out, select it and click **Ungroup.** It is the equivalent of adding parenthesis in a logic expression.

     To adjust the priority of a condition line or a group, select it and click the **Up** or **Down** button.

     To delete a condition line or a group, select it and click the **Delete** button.
3. Click **OK** and the specified filter will be applied to the business view. Only data that meet the filter condition are fetched from the business view and displayed on the data component.

You can also apply a filter to the business view a data component uses in the Query Filter screen of the report wizard while creating the data component.

Notes:

- A JReport Live license for JReport Server is required in order to use this feature. If you do not have the license, contact your Actuate Software account manager to obtain it.
- Business view filters are defined on the component level in Page Report Studio, which means each time you create a component, you can apply a filter to the business view it applies to and it will not affect other components based on the same business view.
- In Page Report Studio, you cannot edit the predefined filters that have been created on a business view at the Designer side. You can just edit the condition based on a predefined filter and then the edited filter will be saved as a user defined filter.

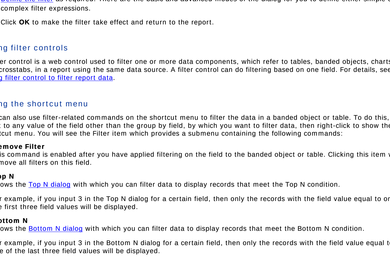
Filtering the data components in a report

When applying filters to the data components themselves in a page report, there are the following ways you can take: using the Filter dialog, using filter controls, using the shortcut menu, and using labels.

Using the Filter dialog

To set the filtering conditions using the Filter dialog:

1. Click **Menu** > **Report** > **Filter**, or the **Filter** button on the Analysis toolbar. The Filter dialog appears.

2. Select the component on which the filtering will be based from the Apply to drop-down list.
3. Define the filter as required. There are the basic and advanced modes of the dialog for you to define either simple or complex filter expressions.
4. Click **OK** to make the filter take effect and return to the report.

Using filter controls

A filter control is a web control used to filter one or more data components, which refer to tables, banded objects, charts, and crosstabs, in a report using the same data source. A filter control can do filtering based on one field. For details, see Using filter control to filter report data.

Using the shortcut menu

You can also use filter-related commands on the shortcut menu to filter the data in a banded object or table. To do this, point to any value of the field other than the group by field, by which you want to filter data, then right-click to show the shortcut menu. You will see the Filter item which provides a submenu containing the following commands:

- **Remove Filter**
  This command is enabled after you have applied filtering on the field to the banded object or table. Clicking this item will remove all filters on this field.
- **Top N**
  Shows the Top N dialog with which you can filter data to display records that meet the Top N condition.

  For example, if you input 3 in the Top N dialog for a certain field, then only the records with the field value equal to one of the first three field values will be displayed.
- **Bottom N**
  Shows the Bottom N dialog with which you can filter data to display records that meet the Bottom N condition.

  For example, if you input 3 in the Bottom N dialog for a certain field, then only the records with the field value equal to one of the last three field values will be displayed.
- **Field values**
  "Field values" is not the name for a command on the Filter submenu, but represents some items which are the values of the field you right-click on. Selecting any field value listed here will make the banded object or table only display records with the field value equal to the selected one.
- **More**
  This command is enabled if the Filter submenu cannot list all field values. When it is enabled, clicking it will show the Select Values dialog. You can select one value in this dialog and apply the setting, after which the banded object or table will only display records with the field value equal to that value.

Using labels

You can also use a label to control the filter condition in a banded object or table. This feature needs to be enabled at report design time.

1. In JReport Designer, select a label in a banded object/table, and then set its Filterable property to true.
2. Set the field by which you want to filter records as the value of the label's Bind Column property.
3. Save the report and publish it to JReport Server.
4. Run the report in Page Report Studio, and you can find that a filter button displayed beside the label. Click it to show the Filter list, which contains All, Top N, Bottom N, Custom Filter, the field values, and More (if there are too many distinct values for the field), then click the corresponding item to filter the records.

   After a filter is applied on the field specified by the Bind Column property, the button will be affixed with a check mark. You can still click it to show the Filter list to change the filter condition or select the All item to remove the filter on the field.

Notes:

- You can also filter records by using the shortcut menu for a label in the same way as for a field value, provided you have set its Bind Column property value to a field.
- When using the shortcut menu for a field value or label to filter, all the items (Remove Filter, Top N, Bottom N, and More) will be showed by default. If you want to disable some of the items, you should set the field's Filter Options property when designing the report in JReport Designer.
- When filtering the data using either shortcut menu or labels, you may notice that the corresponding filter expression displays in the Filter dialog if you open this dialog.
- JReport allows you to define the display names for fields to be shown in the Filter dialog.
- For page reports created in JReport Designer, the default selected component and available components in the Apply to drop-down list of the Filter dialog are determined by the Default for Filter and Invisible for Filter Dialogs properties of the components in the page reports.

Using view elements

After a report has been built and published to JReport Server, you can open it in Page Report Studio and use the Resource View panel to analyze data of the report by dragging view elements from the panel to the component (banded object, table or crosstab) in the report, provided that the data objects used by the component can be converted to corresponding view elements (see the note in Analytic reporting for details).

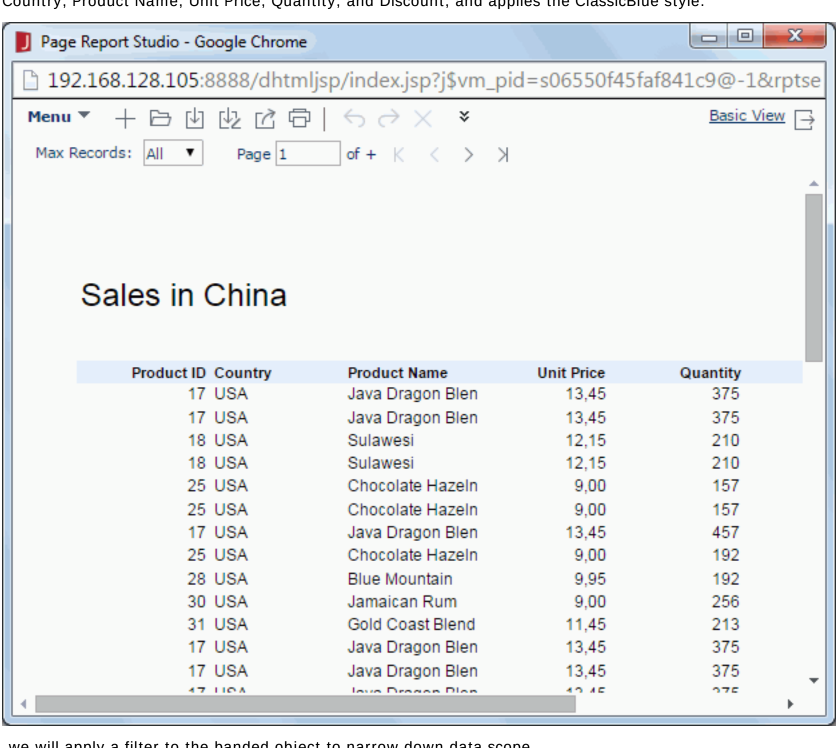
Tip: To display the Resource View panel, click **Menu** > **View** > **Resource View** or the **Resource View** button on the View toolbar. You can use the search bar at the top of the panel to search for any desired resource in a fast and convenient way.

The following examples show how to analyze reports using view elements. These examples are based on the business view WorldWideSalesBV in Data Source 1 of the SampleReports catalog.

- Example 1: Analyzing a banded report
- Example 2: Analyzing a crosstab report
- Example 3: Analyzing a table report

Example 1: Analyzing a banded report

1. In Page Report Studio, design a banded report titled Sales in China on WorldWideSalesBV, which shows the fields Product ID, Country, Product Name, Unit Price, Quantity, and Discount, and applies the ClassicBlue style.

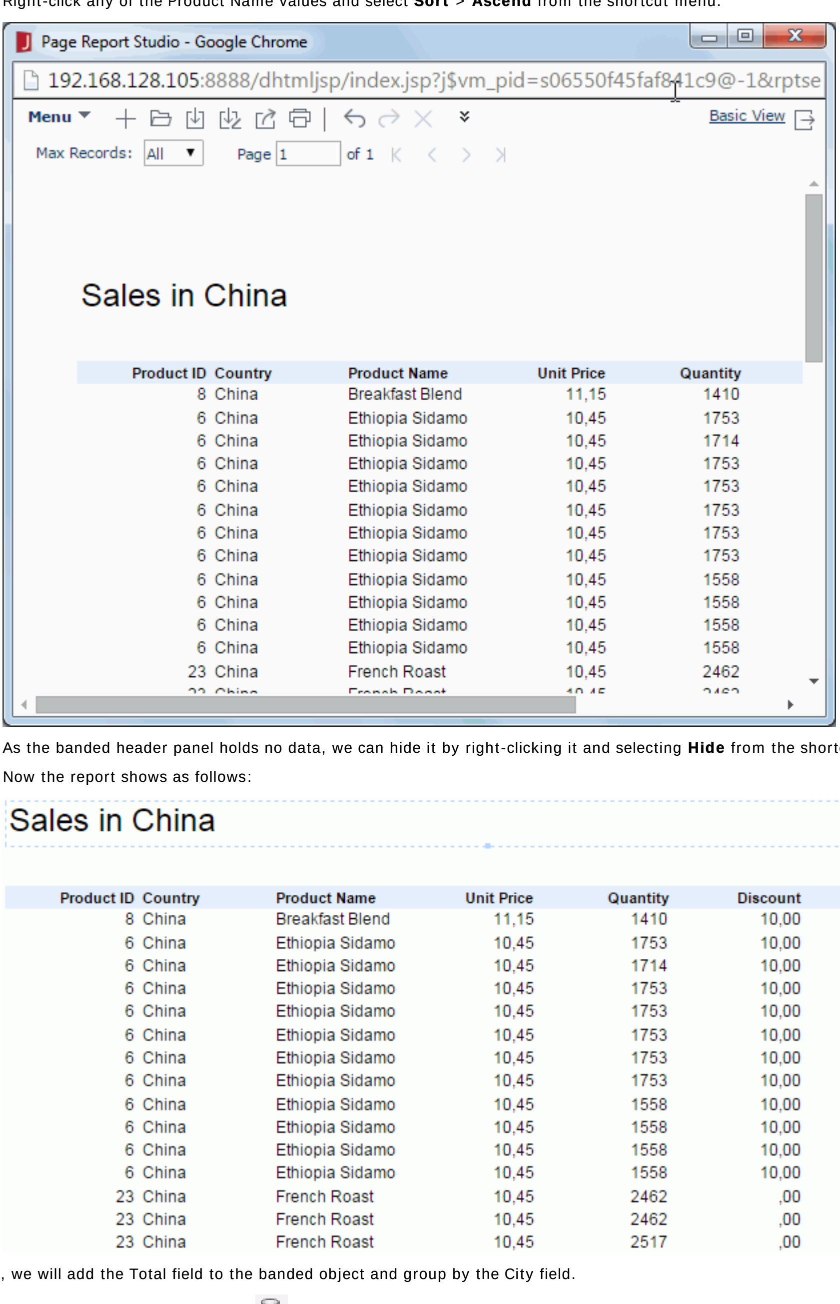


First, we will apply a filter to the banded object to narrow down data scope.

2. Click the **Filter** button on the Analysis toolbar. In the Filter dialog, define the filter as COUNTRY = 'China'.

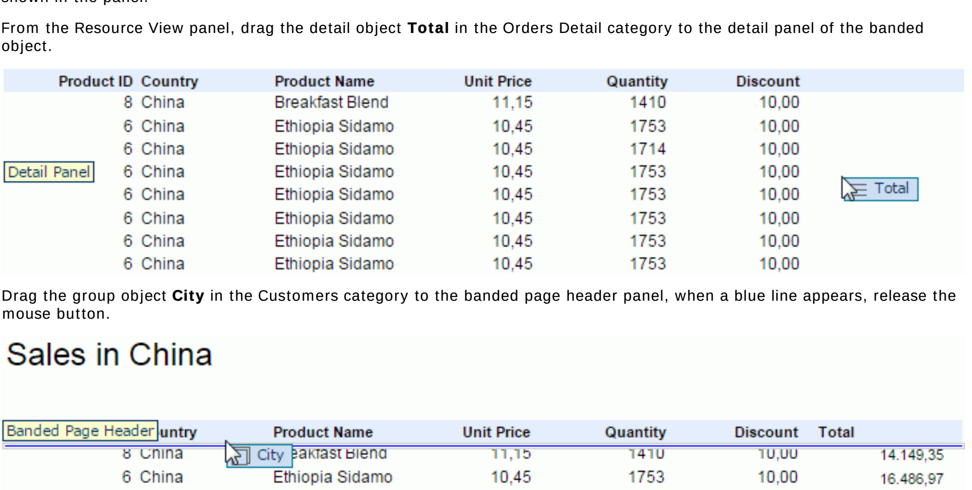
We want to further sort the banded object by Product Name ascending.

3. Right-click any of the Product Name values and select **Sort** > **Ascend** from the shortcut menu.



4. As the banded header panel holds no data, we can hide it by right-clicking it and selecting **Hide** from the shortcut menu.

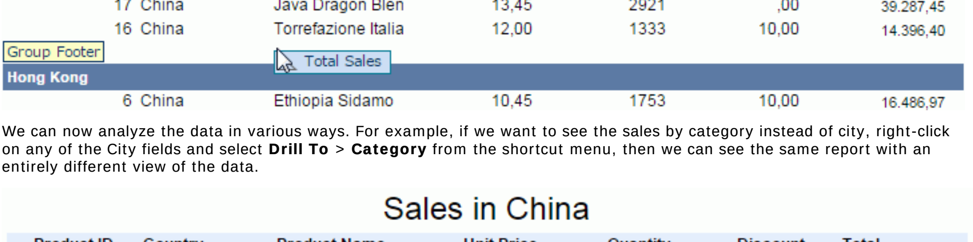
   Now the report shows as follows:



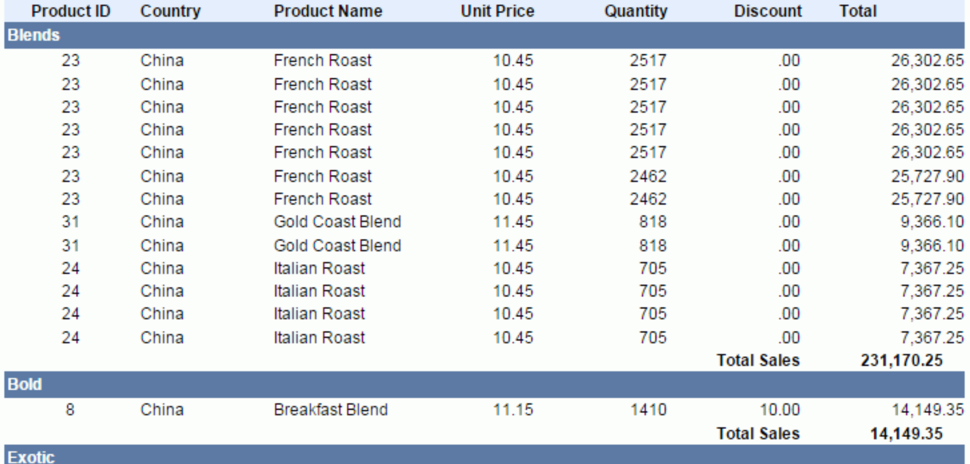
Next, we will add the Total field to the banded object and group by the City field.

5. Click the **Resource View** button on the View toolbar, then resources of the business view the banded object uses will be shown in the panel.
6. From the Resource View panel, drag the detail object **Total** in the Orders Detail category to the detail panel of the banded object.

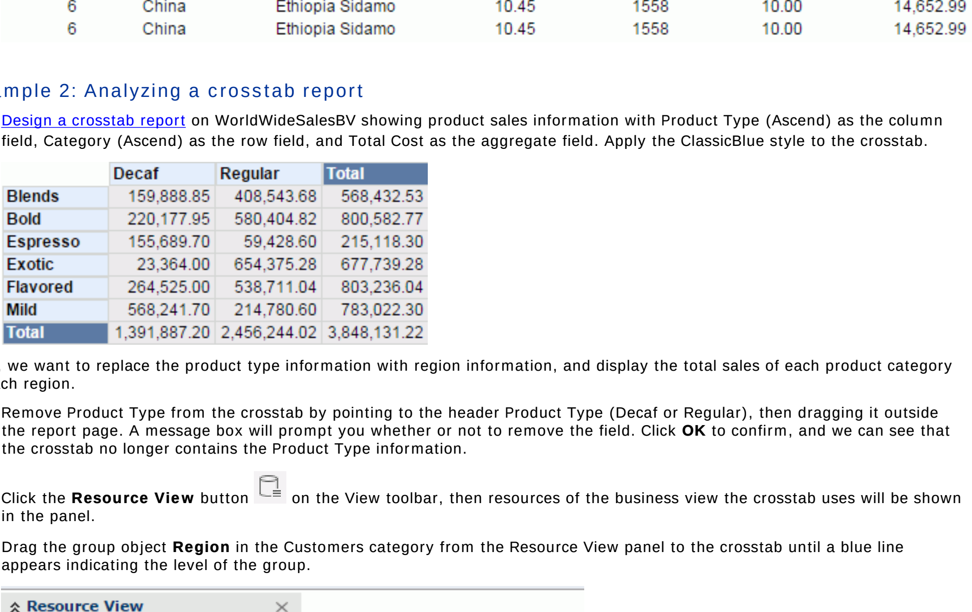
7. Drag the group object **City** in the Customers category to the banded page header panel, when a blue line appears, release the mouse button.



8. Finally, drag the aggregation object **Total Sales** in the Orders Detail category to the group footer panel.

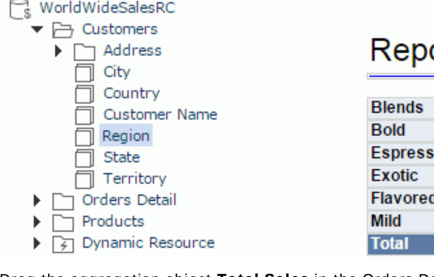


9. We can now analyze the data in various ways. For example, if we want to see the sales by category instead of city, right-click on any of the City fields and select **Drill To** > **Category** from the shortcut menu, then we can see the same report with an entirely different view of the data.



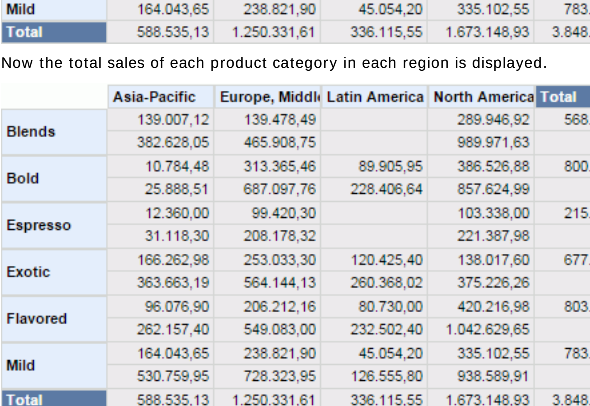
Example 2: Analyzing a crosstab report

1. Design a crosstab report on WorldWideSalesBV showing product sales information with Product Type (Ascend) as the column field, Category (Ascend) as the row field, and Total Cost as the aggregate field. Apply the ClassicBlue style to the crosstab.

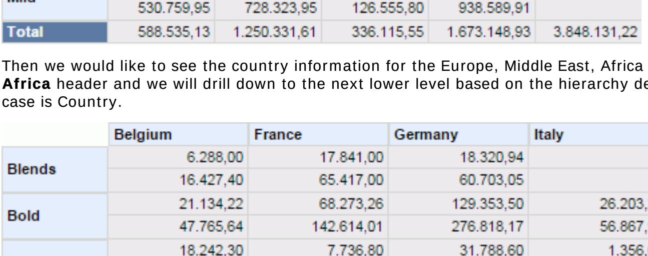


First, we want to replace the product type information with region information, and display the total sales of each product category in each region.

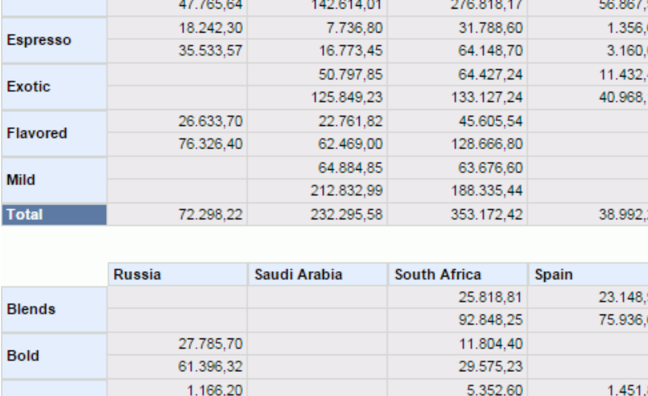
2. Remove Product Type from the crosstab by pointing to the header Product Type (Decaf or Regular), then dragging it outside the report page. A message box will prompt you whether or not to remove the field. Click **OK** to confirm, and we can see that the crosstab no longer contains the Product Type information.
3. Click the **Resource View** button on the View toolbar, then resources of the business view the crosstab uses will be shown in the panel.
4. Drag the group object **Region** in the Customers category from the Resource View panel to the crosstab until a blue line appears indicating the level of the group.



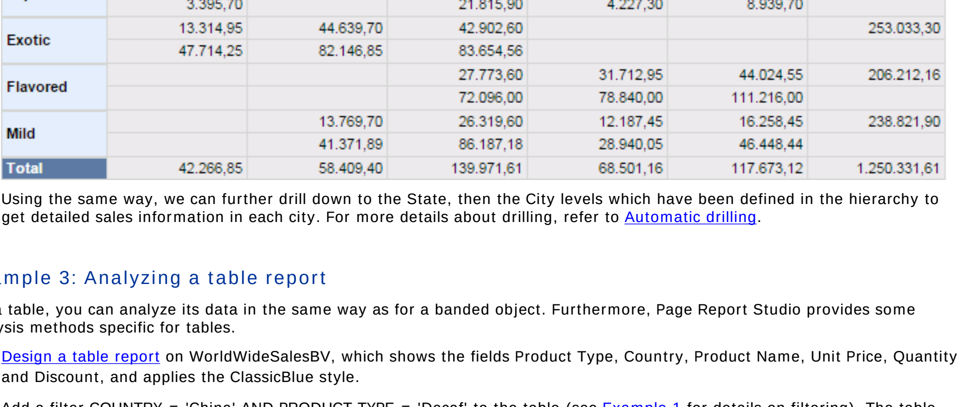
5. Drag the aggregation object **Total Sales** in the Orders Detail category to the aggregate area of the crosstab.

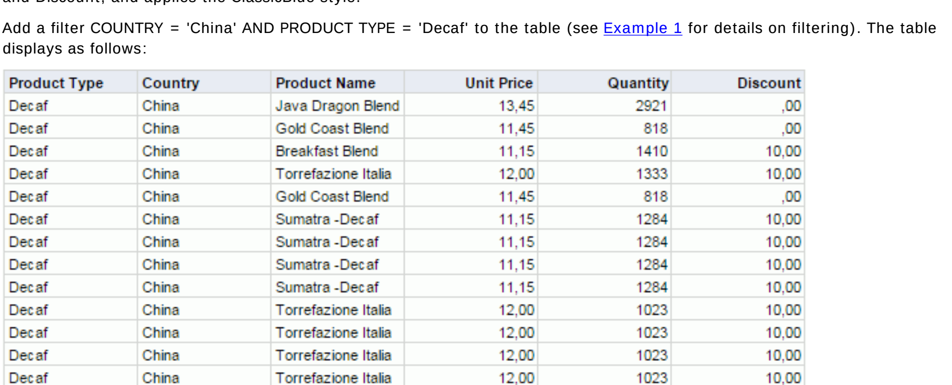


   Now the total sales of each product category in each region is displayed.



6. Then we would like to see the country information for the Europe, Middle East, Africa region. Click in the **Europe, Middle East, Africa** header and we will drill down to the next lower level based on the hierarchy defined in the business view which in this case is Country.



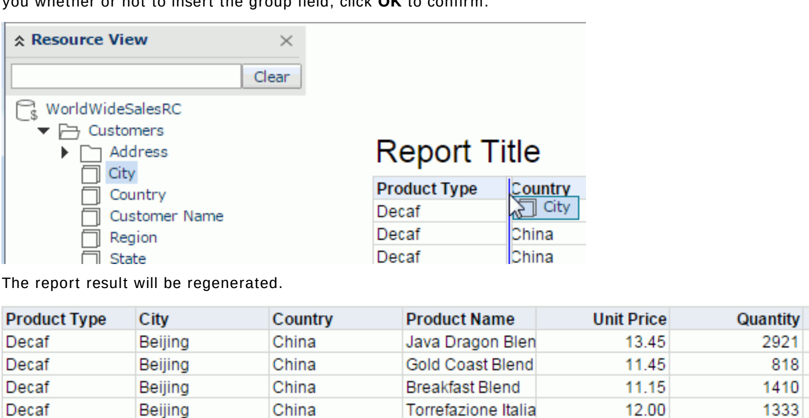


   Using the same way, we can further drill down to the State, then the City levels which have been defined in the hierarchy to get detailed sales information in each city. For more details about drilling, refer to Automatic drilling.

Example 3: Analyzing a table report

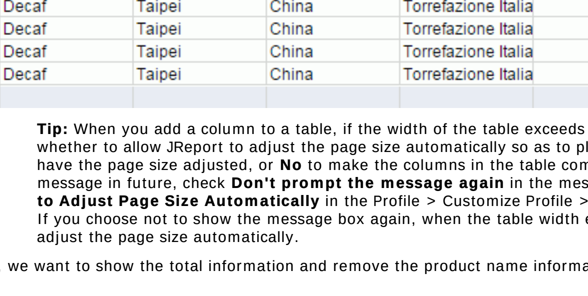
For a table, you can analyze its data in the same way as for a banded object. Furthermore, Page Report Studio provides some analysis methods specific for tables.

1. Design a table report on WorldWideSalesBV, which shows the fields Product Type, Country, Product Name, Unit Price, Quantity and Discount, and applies the ClassicBlue style.
2. Add a filter COUNTRY = 'China' AND PRODUCT TYPE = 'Decaf' to the table (see Example 1 for details on filtering). The table displays as follows:



For a table, we can insert a column (or row for horizontal table) at a specific position. So next, we will insert the group object City into the table.

3. Click the **Resource View** button on the View toolbar, then resources of the business view the table uses will be shown in the panel.
4. Drag **City** in the Customers category from the Resource View panel to the boundary between the first column (Product Type) and the second column (Country) in the table until a blue line appears. Release the mouse button and a message will prompt you whether or not to insert the group field, click **OK** to confirm.

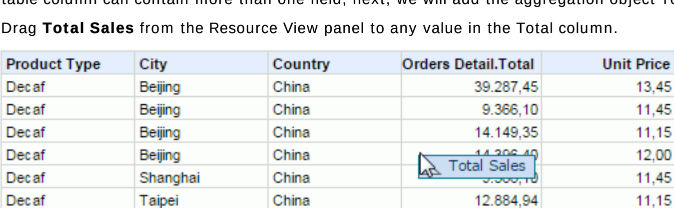


   The report result will be regenerated.

   Tip: When you add a column to a table, if the width of the table exceeds the defined page size, you will be prompted whether to allow JReport to adjust the page size automatically so as to place the column. Click **Yes** in the message box to have the page size adjusted, or **No** to make the columns in the table compressed. Also, If you do not want to display the message in future, check **Don't prompt the message again** in the message box, or uncheck **Always Prompt Whether to Adjust Page Size Automatically** in the Profile > Customize Profile > Page Report Studio > Properties > Default tab. If you choose not to show the message box again, when the table width exceeds the defined page size, JReport will always adjust the page size automatically.

Next, we want to show the total information and remove the product name information. This can be done with a single drag-and-drop.

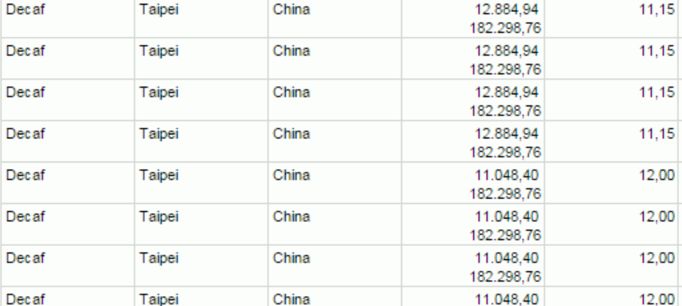
5. Drag the detail object **Total** in the Orders Detail category to the header Product Name until the label Product Name is highlighted in a blue background.



   Now, the total value for each record will be generated.

As a table column can contain more than one field, next, we will add the aggregation object Total Sales to the Total column.

6. Drag **Total Sales** from the Resource View panel to any value in the Total column.

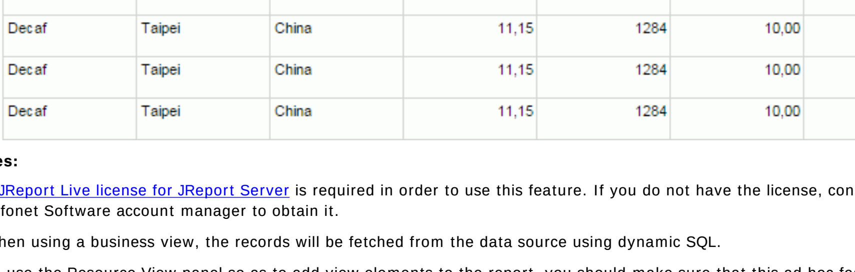


   The report result will be regenerated.

   Here 182,298.76 is the sum of all total values. In this way, the title for the added field will not be automatically created.

At last, we want to change the order of the Total and Discount columns in the table.

7. Drag the label **Total** to the right of the Discount column, when a blue line appears along the right boundary of the Discount column, release the mouse button.



   We can see that order of the columns changes.

Notes:

- A JReport Live license for JReport Server is required in order to use this feature. If you do not have the license, contact your Jinfonet Software account manager to obtain it.
- When using a business view, the records will be fetched from the data source using dynamic SQL.
- To use the Resource View panel so as to add view elements to the report, you should make sure that this ad hoc feature is enabled in the specified Page Report Studio feature profile. This setting can only be made by administrators.

Using dynamic resources

When you drag view elements from the Resource View panel to analyze data of a report, sometimes you may find that the view elements that have been predefined in the business view cannot meet your requirements, in which case, you can create some dynamic resources and use them in the report to get the desired data. Then when you save the report, the dynamic resources will be saved along with the report as its resources.

Dynamic resources are report tab level resources, which means they are only available to the report tab for which they are created.

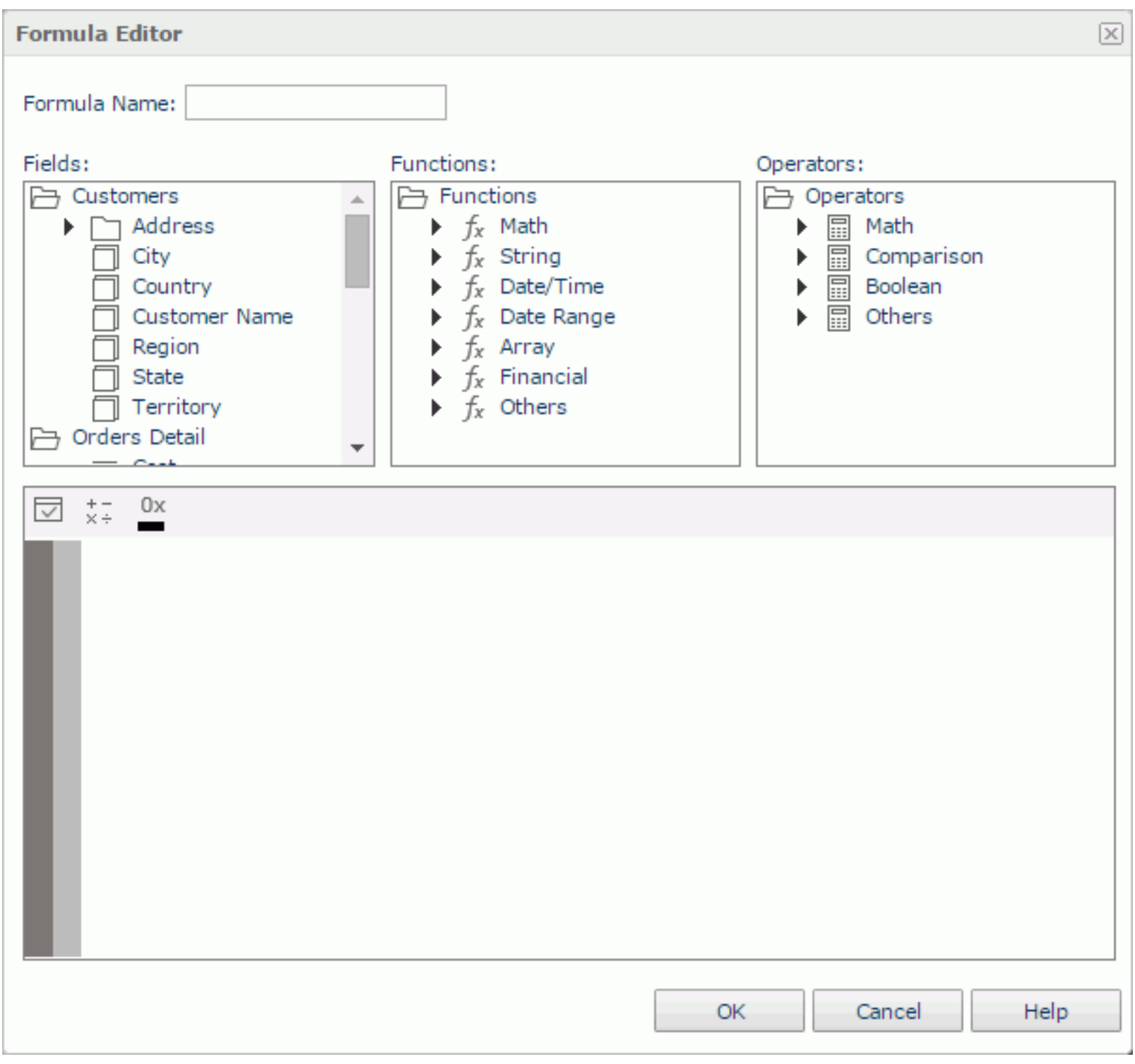
Dynamic resources in Page Report Studio include formulas and Aggregations.

Creating and using dynamic formulas

You should have some knowledge of the formula syntax before you can successfully compose a formula with no errors. To learn the formula syntax, refer to Formula syntax.

To create a dynamic formula:

1. In the Resource View panel, expand the **Dynamic Resources** > **Formulas** node, then click **<Add Formula•gt;**. The Formula Editor appears.



2. Enter a name for the formula in Formula Name text field.
3. Compose the formula by selecting the required fields, functions and operators from the Fields, Functions and Operators boxes. You can also write the formula by yourself in the editing box.

   For details about the functions and operators, refer to Built-in functions and Operators.

4. Click the **Check** button to check whether or not the syntax of your formula is correct.
5. When done, click the **OK** button to create the formula.

Once a dynamic formula is created, you can then drag it from the Resource View panel to the desired position in the report as a detail object for data analyzing. The formulas can also be used to control object properties if you are an advanced user and provided that the Use Dynamic Formula in Property is checked on the Report menu.

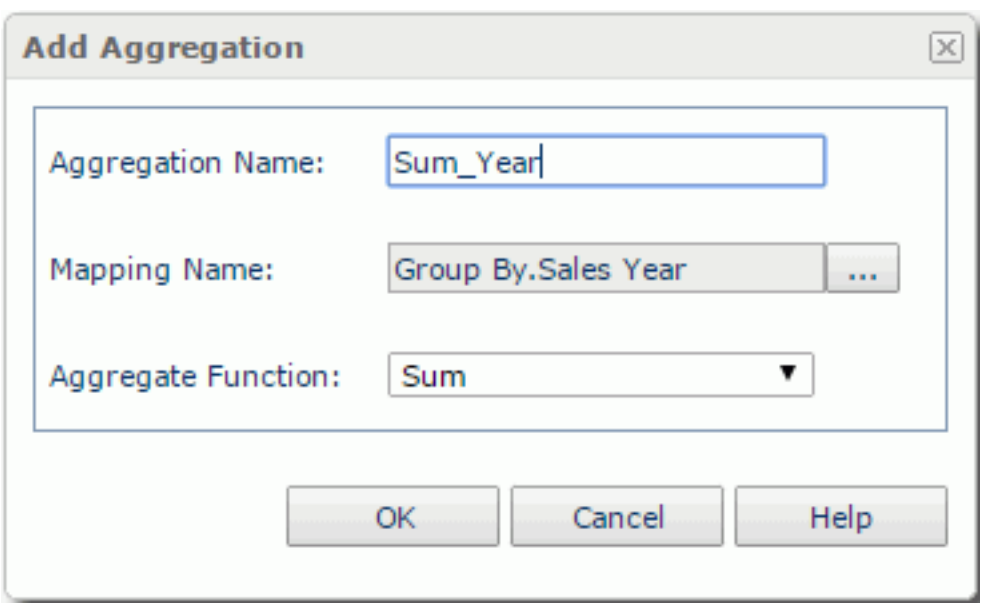
If you want to further edit an existing formula or remove any formula that is not required, right-click the formula and then click the corresponding command on the shortcut menu. However, if the formula has been used in the report or referenced by another formula, it cannot be deleted.

Creating and using dynamic aggregation objects

In Page Report Studio, you can also create dynamic aggregation objects by mapping them to the available resources which include group objects, detail objects in the current business view and the dynamic formulas that have been created in the report.

To create a dynamic aggregation object:

1. In the Resource View panel, expand the **Dynamic Resources** > **Aggregations** node, then click **<Add Aggregation•gt;**. The Add Aggregation dialog appears.



2. In the Aggregation Name text field, specify the display name of the aggregation object.
3. Click the button ... next to the Mapping Name text field. In the Select Source dialog, specify a field or a formula on which the aggregation object is based.
4. From the Aggregate Function drop-down list, specify the aggregate function for the aggregation object.
5. When done, click **OK** to create the aggregation object.

You can also create a dynamic aggregation object on a dynamic formula. To do this:

1. In the Resource View panel, right-click the formula in the Dynamic Resources > Formulas node, then select **Add Aggregation** from the shortcut menu.
2. In the Add Aggregation dialog, specify the display name of the aggregation object and the aggregate function as required.
3. When done, click **OK**.

Once a dynamic aggregation object is created, you can then drag it from the Resource View panel to the desired position in the report so as the get the desired data.

For any dynamic aggregation object, you can further modify if needed. To do this, right-click the aggregation object and select **Edit** from the shortcut menu. Then in the Edit Aggregation dialog, edit the aggregation object as required.

For dynamic aggregation objects that are on longer required, you can remove them. To delete an aggregation object, right-click it and click **Delete** on the shortcut menu. Aggregation objects that have been used in report cannot be deleted.

Notes:

- A JReport Live license for JReport Server is required in order to use this feature. If you do not have the license, contact your Jinfonet Software account manager to obtain it.
- You can only use JDK (not JRE) to compile formulas created in JReport and save a dynamic formula with no errors into a report.
- Currently, global variables are not supported in dynamic formulas.
- When formulas reference display names or mapping names, the names should not contain any of below characters if the names are not quoted by double-quotation marks "":

  "~", "`", "!", "@", "#", "$", "%", "^", "&", "*", "(", ")", "-", "+", "=", "{", "}", "[", "]", "|", "\\", ":", ";", "\", " ' ", "<", ",", ">", ".", "?", "/"

  Examples:

  - Expression @Customer#; will cause a syntax error. But @"Customer#" is ok.
  - If a field has the display name Category.Measure, when adding it to a formula, quote it as "Category. Measure" or "Category"."Measure".
- Now in JReport, the display names of objects in a category in a business view cannot be duplicated. When you choose to create a dynamic formula/aggregation object on an object which was created in a previous version and it has the same display name as another object, you will be prompted with a message asking you to give a new name for the object in JReport Designer first.

Drilling through the report data

In a page report runing in Page Report Studio, you can choose to show certain groups of records according to your requirements, and switch among the groups to see the data you want.

This section presents two kinds of drilling in Page Report Studio. They are:

- Automatic drilling
- Going

Automatic drilling

Automatic Drilling enables you to switch from the current group to another group by using system-defined commands on the shortcut menu, and it is divided into four kinds:

- Drill-to
  It enables you to obtain a different view of data by switching among groups.
- Drill-to-by-value
  It enables you to filter data based on a drill-to action so as to obtain a more detailed view of the data.
- Drill-down
  It enables you to drill data to lower groups according to predefined hierarchies.
- Drill-up
  It enables you to drill data to higher groups according to predefined hierarchies.

Drilling actions are performed on crosstabs, charts, grouped tables and banded objects, whose data are based on business view or if on query, data fields of which can be converted to corresponding view elements (see the note in Analytic reporting for details). After drilling, the new component can be analyzed in the same way as the original one.

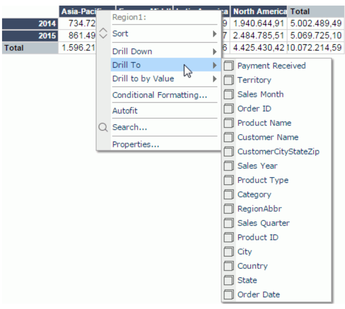
Assume you have created a crosstab report on the business view WorldWideSalesBV in Data Source 1 of the SampleReports catalog showing product sales information with Region as the column field, Sales Year as the row field, and Total Sales as the summary field, and applied the Neutral style to the crosstab. The crosstab shows as follows:

	Asia-Pacific	Europe, Middl	Latin America	North America	Total
2014	734.722,78	1.960.140,31	366.981,49	1.940.644,91	5.002.489,49
2015	861.492,62	1.242.595,60	480.851,37	2.484.785,51	5.069.725,10
Total	1.596.215,39	3.202.735,92	847.832,85	4.425.430,42	10.072.214,59

We will now take the crosstab as an instance to illustrate the automatic drilling functions.

Drill-to

1. Right-click any value of Region, Asia-Pacificfor example, and choose **Drill To** from the shortcut menu. The list of groups available for Drill To will appear on the submenu.



2. Click **Product Type** on the submenu, then in the regenerated result, we can see that Sales Year remains the group for rows and Product Type becomes the group for columns.

	Decaf	Regular	Total
2014	2.073.048,97	2.929.440,51	5.002.489,49
2015	1.958.999,48	3.110.725,62	5.069.725,10
Total	4.032.048,45	6.040.166,13	10.072.214,59

3. Repeat Steps 1 and 2 to drill the data to other groups. Row field can also be drilled freely.
4. To go back to the original report, right-click any value of Product Type, choose **Drill To** > **Region** from the shortcut menu.

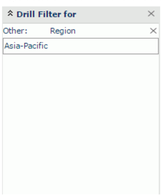
Drill-to-by-value

1. Go back to the original report in the above example.
2. Right-click the value **Asia-Pacific** of the Region group, and point to **Drill to by Value** on the shortcut menu. A submenu for the command is displayed, which lists the same items as those of Drill To.
3. Click **Product Type** too and the result will be regenerated.

	Decaf	Regular	Total
2014	302.284,22	432.438,55	734.722,78
2015	398.056,94	463.435,68	861.492,62
Total	700.341,16	895.874,23	1.596.215,39

We can see that the result is different from that of drill-to. This is because that, for the drill-to-by-value action, the group of columns changes to Product Type by the Region value Asia-Pacific. That is, on the basis of the drill-to action, a filtering action where Region = Asia-Pacific is further performed, and thus the result of drill-to-by-value is generated.

In addition, when a drill-to-by-value action is performed, the Drill Filter panel will be displayed on the left of the Page Report Studio window, which shows the group and the value the filter is based on.

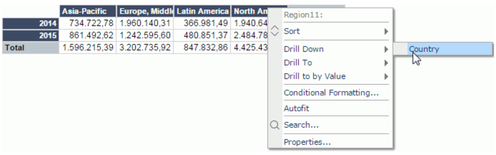


4. To go back to the original report, first delete the drill filter in the Drill Filter panel by clicking **X** next to the group name, then right-click any value of Product Type and click **Drill To** > **Region** from the shortcut menu.

Drill-down

Drill-down actions are based on predefined business view hierarchies. The WorldWideSalesBV contains a hierarchy Geography, which allows you to drill a group (corresponding to a high level) down to the one-level-lower group.

1. Go back to the original report in the above example, right-click the value **North America** , on the shortcut menu, point to **Drill Down**, and we can see that Country is listed as the submenu item.



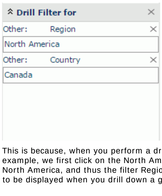
2. Click **Country** to see the result. It displays the data about countries in the North America region.

	Canada	Mexico	USA	Total
2014	640.390,80		1.300.254,10	1.940.644,91
2015	235.126,99	912.729,91	1.336.928,61	2.484.785,51
Total	875.517,80	912.729,91	2.637.182,72	4.425.430,42

3. The one-level-lower group for Country defined in the hierarchy is State. Now click **Canada** directly and JReport will also drill it down to State.

	BC	ON	Total
2014	549.082,76	91.308,04	640.390,80
2015	235.126,99		235.126,99
Total	784.209,75	91.308,04	875.517,80

After these two drill-down actions, we can see two filters are added in the Drill Filter panel, Region = North America and Country = Canada.

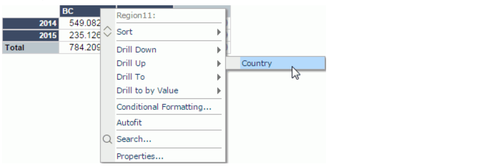


This is because, when you perform a drill-down action, a filter will be created based on the value you click on. In this example, we first click on the North Americaregion, so JReport drills this region one-level down to display countries in North America, and thus the filter Region = North America is created. If you want all data in the one-level-lower group to be displayed when you drill down a group, you can remove the corresponding filter from the Drill Filter panel.

Drill-up

Drill-up actions allow you to drill a group (corresponding to a low level) up to the one-level-higher group.

1. Based on the report result after drill-down, right-click any value of State, BC for example, on the shortcut menu, point to **Drill Up**, and we can see that Country is listed as the submenu item.



2. Click **Country** to see the result. The group is drilled one level up. Country is now the group for columns.

	Canada	Mexico	USA	Total
2014	640.390,80		1.300.254,10	1.940.644,91
2015	235.126,99	912.729,91	1.336.928,61	2.484.785,51
Total	875.517,80	912.729,91	2.637.182,72	4.425.430,42

3. The one-level-higher group for Country defined in the hierarchy is Region. Now right-click any value of Country, Canada for example, on the shortcut menu, point to **Drill Up** and click **Region**, JReport will drill it up to Region.

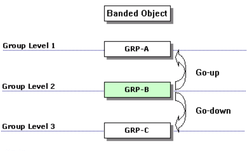
	Asia-Pacific	Europe, Middl	Latin America	North America	Total
2014	734.722,78	1.960.140,31	366.981,49	1.940.644,91	5.002.489,49
2015	861.492,62	1.242.595,60	480.851,37	2.484.785,51	5.069.725,10
Total	1.596.215,39	3.202.735,92	847.832,86	4.425.430,42	10.072.214,59

Notes:

- A JReport Live license for JReport Server is required in order to use this feature. If you do not have the license, contact your Jinfonet Software account manager to obtain it.
- For banded object and table, you can right-click its group header/footer to show the shortcut menu so as to use the automatic drilling functions, you can also right-click field values in the group header/footer to achieve this.
- For group objects that not used as group fields in a banded object or table, automatic drilling doesn't take effect.
- JReport allows you to define display names for items to be shown on the drill-related menu items.
- You can also directly click group objects to perform the drill-down action, however, for page reports created in JReport Designer, if there are other actions defined on the group objects at the same time, you can drill down the objects by a single click only when the click priority of the drill-down action is specified to be the highest at report design time. Otherwise, you have to use the Drill Down command on the objects' shortcut menu to perform the action.

Going

In a page report running in Page Report Studio, you can select to show certain groups of records in a banded object according to your requirements. You can also switch among the groups to see the data you want. This action is called going, which divides into go-to, go-up, go-down, and go-to-detail, as indicated in the diagram.



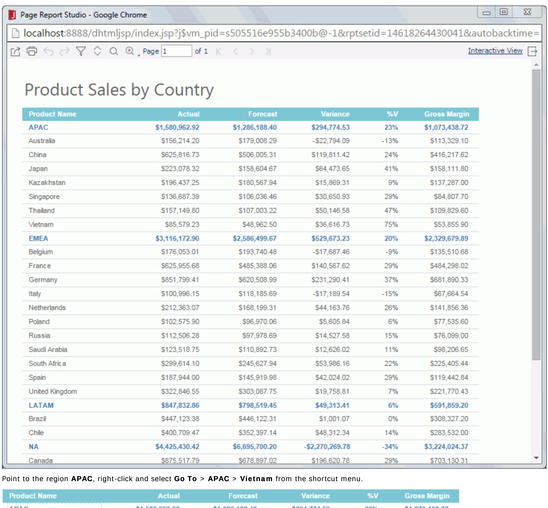
- Go-to
  The go-to action allows you to switch the data presented in a banded object from any group to any other group.
- Go-up
  Go-up means to jump up one group level to show the records of a particular group.
- Go-down
  Go-down means to jump down one group level to show the records of a particular group.
- Go-to-detail
  Go-to-detail allows you to concentrate on the details of a group.

Going actions are available only for banded objects that contain groups, and fields in which have not been converted to view elements of a business view. Going actions do not apply to banded objects created in Page Report Studio. After a going action has been performed, the data presented in the banded object will be re-loaded from the data buffer, showing only the records in the selected group, and the new report created by going can also be viewed, printed, and exported to other format in the same way as the original report. In addition, a "going path", which tracks the going action, will be displayed in the Go To drop-down list on the navigation bar, with which you can easily return to the original report.

The following describes the use of the going actions based on Banded_Link.cls in the SampleReports folder of Public Reports, which contains a banded report.

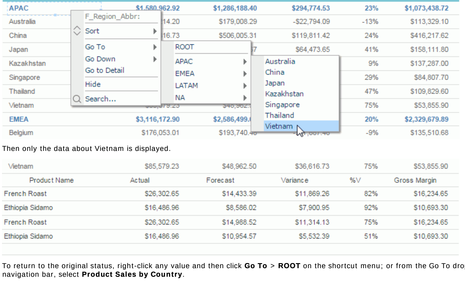
Go-to

1. Run Banded_Link.cls.



Product Name	Actual	Forecast	Variance	%V	Gross Margin
APAC	$1,580,962.92	$1,286,188.40	$294,774.53	23%	$1,073,438.72
Australia	$156,214.20	$179,008.29	-$22,794.09	-13%	$113,329.10
China	$625,816.73	$506,005.31	$119,811.42	24%	$416,217.62
Japan	$223,078.32	$158,604.67	$64,473.65	41%	$158,111.80
Kazakhstan	$196,437.25	$180,567.94	$15,869.31	9%	$137,287.00
Singapore	$136,687.39	$106,036.46	$30,650.93	29%	$84,807.70
Thailand	$157,149.80	$107,003.22	$50,146.58	47%	$109,829.60
Vietnam	$85,579.23	$48,962.50	$36,616.73	75%	$53,855.90
EMEA	$3,116,172.90	$2,586,499.67	$529,673.23	20%	$2,329,679.89
Belgium	$176,053.01	$193,740.48	-$17,687.46	-9%	$135,510.68
France	$625,955.68	$485,388.06	$140,567.62	29%	$484,298.02
Germany	$851,799.41	$620,508.99	$231,290.41	37%	$681,890.33
Italy	$100,996.15	$118,185.69	-$17,189.54	-15%	$67,664.54
Netherlands	$212,363.07	$168,199.31	$44,163.76	26%	$141,856.36
Poland	$102,575.90	$96,970.06	$5,605.84	6%	$77,535.60
Russia	$112,506.28	$97,978.69	$14,527.58	15%	$76,099.00
Saudi Arabia	$123,518.75	$110,892.73	$12,626.02	11%	$98,206.65
South Africa	$299,614.10	$245,627.94	$53,986.16	22%	$225,405.44
Spain	$187,944.00	$145,919.98	$42,024.02	29%	$119,442.84
United Kingdom	$322,846.55	$303,087.75	$19,758.81	7%	$221,770.43
LATAM	$847,832.86	$798,519.45	$49,313.41	6%	$591,859.20
Brazil	$447,123.38	$446,122.31	$1,001.07	0%	$308,327.20
Chile	$400,709.47	$352,397.14	$48,312.34	14%	$283,532.00
NA	$4,425,430.42	$6,695,700.20	-$2,270,269.78	-34%	$3,224,024.37
Canada	$875,517.79	$678,897.02	$196,620.78	29%	$703,130.31

2. Point to the region **APAC**, right-click and select **Go To** > **APAC** > **Vietnam** from the shortcut menu.



Then only the data about Vietnam is displayed.

Product Name	Actual	Forecast	Variance	%V	Gross Margin
Vietnam	$85,579.23	$48,962.50	$36,616.73	75%	$53,855.90
Product Name	Actual	Forecast	Variance	%V	Gross Margin
French Roast	$26,302.65	$14,433.39	$11,869.26	82%	$16,234.65
Ethiopia Sidamo	$16,486.96	$8,586.02	$7,900.95	92%	$10,693.30
French Roast	$26,302.65	$14,988.52	$11,314.13	75%	$16,234.65
Ethiopia Sidamo	$16,486.96	$10,954.57	$5,532.39	51%	$10,693.30

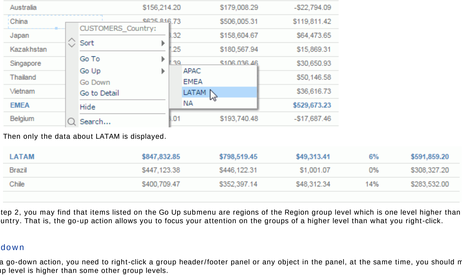
3. To return to the original status, right-click any value and then click **Go To** > **ROOT** on the shortcut menu; or from the Go To drop-down list on the navigation bar, select **Product Sales by Country**.

You may notice that the result is not dependent on what you right-clicked, in other words, you can right-click any field value in the banded object or even the blank part of a group header/footer panel or detail panel, in order to perform a go-to action.

Go-up

For a go-up action, you need to right-click a group header/footer panel or any object in the panel, at the same time, you should make sure that this group level is lower than some other group levels.

1. Undo the go-to action in the above example.
2. Point to any country, for example China, right-click and select **Go Up** > **LATAM** from the shortcut menu.



Then only the data about LATAM is displayed.

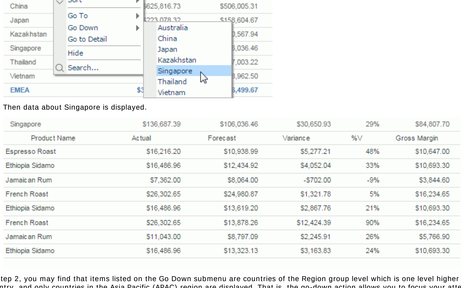
LATAM	$847,832.85	$798,519.45	$49,313.41	6%	$591,859.20
Brazil	$447,123.38	$446,122.31	$1,001.07	0%	$308,327.20
Chile	$400,709.47	$352,397.14	$48,312.34	14%	$283,532.00

At Step 2, you may find that items listed on the Go Up submenu are regions of the Region group level which is one level higher than the current group level - Country. That is, the go-up action allows you to focus your attention on the groups of a higher level than what you right-click.

Go-down

For a go-down action, you need to right-click a group header/footer panel or any object in the panel, at the same time, you should make sure that this group level is higher than some other group levels.

1. Undo the go-up action in the above example.
2. Point to **APAC**, right-click and select **Go Down** > **Singapore** from the shortcut menu.



Then data about Singapore is displayed.

Singapore	$136,687.39	$106,036.46	$30,650.93	29%	$84,807.70
Product Name	Actual	Forecast	Variance	%V	Gross Margin
Espresso Roast	$16,216.20	$10,938.99	$5,277.21	48%	$10,647.00
Ethiopia Sidamo	$16,486.96	$12,434.92	$4,052.04	33%	$10,693.30
Jamaican Rum	$7,362.00	$8,064.00	-$702.00	-9%	$3,844.60
French Roast	$26,302.65	$24,960.87	$1,321.78	5%	$16,234.65
Ethiopia Sidamo	$16,486.96	$13,619.20	$2,867.76	21%	$10,693.30
French Roast	$26,302.65	$13,878.26	$12,424.39	90%	$16,234.65
Jamaican Rum	$11,043.00	$8,797.09	$2,245.91	26%	$5,766.90
Ethiopia Sidamo	$16,486.96	$13,323.13	$3,163.83	24%	$10,693.30

At Step 2, you may find that items listed on the Go Down submenu are countries of the Region group level which is one level higher than the group level of Country, and only countries in the Asia Pacific (APAC) region are displayed. That is, the go-down action allows you to focus your attention on the groups of a lower level than what you right-click, and only those lower-level groups which are related with the higher-level group value you right-click will be concerned.

Go-to-detail

If a banded object contains group information, then a field, label, image or shape map in a group header/footer panel of the banded object can be used to obtain information of that group, and a chart in a banded object also has the similar function. The go-to-detail action should be predefined at report design time.

1. Undo the go-down action in the above example.
2. Point to **LATAM**, right-click and select **Go to Detail** from the shortcut menu. Then only the data about this region is displayed. You can also click LATAM directly to perform the go-to-detail action, provided that the click priority of the action is specified to be the highest at report design time.

LATAM	$847,832.85	$798,519.45	$49,313.41	6%	$591,859.20
Brazil	$447,123.38	$446,122.31	$1,001.07	0%	$308,327.20
Product Name	Actual	Forecast	Variance	%V	Gross Margin
Java Dragon Blend	$38,426.65	$53,953.71	-$15,527.06	-29%	$26,855.80
Sulawesi	$19,440.00	$32,978.57	-$13,538.57	-41%	$6,640.00
Java Dragon Blend	$68,689.15	$64,070.03	$4,619.12	7%	$48,005.80
Ethiopia Sidamo	$28,929.78	$26,871.43	$2,058.35	8%	$18,763.60
Breakfast Blend	$25,378.51	$22,937.14	$2,441.37	11%	$18,208.80
Torrefazione Italia	$25,833.60	$30,171.43	-$4,337.83	-14%	$19,734.00
Ethiopia Sidamo	$28,929.78	$24,482.86	$4,446.92	18%	$18,763.60
Breakfast Blend	$25,378.51	$21,025.71	$4,352.80	21%	$18,208.80
Torrefazione Italia	$25,833.60	$28,800.00	-$2,966.40	-10%	$19,734.00
Ethiopia Sidamo	$28,929.78	$19,705.71	$9,224.07	47%	$18,763.60
Breakfast Blend	$25,378.51	$21,025.71	$4,352.80	21%	$18,208.80
Torrefazione Italia	$25,833.60	$19,885.71	$5,947.89	30%	$19,734.00
Ethiopia Sidamo	$28,929.78	$32,245.71	-$3,315.93	-10%	$18,763.60
Breakfast Blend	$25,378.51	$27,397.14	-$2,018.63	-7%	$18,208.80
Torrefazione Italia	$25,833.60	$20,571.43	$5,262.17	26%	$19,734.00
Chile	$400,709.47	$352,397.14	$48,312.34	14%	$283,532.00
Product Name	Actual	Forecast	Variance	%V	Gross Margin
Ethiopia Sidamo	$28,929.78	$30,454.29	-$1,524.51	-5%	$18,763.60

Manipulating data components

You can manipulate data components, which refer to crosstabs, tables, banded objects, charts and geographic maps, in Page Report Studio as shown below. However, most of the manipulations require selecting the component first. To select a component, click anywhere in the component, when the icon appears at its upper left corner, click the icon.

Note: When manipulating data components, a JReport Live license for JReport Server is required in order to use the features involving business view or changes of report template. If you do not have the license, contact your Jinfonet Software account manager to obtain it.

Setting the number of records retrieved by data components

You can set the number of records that can be retrieved by all data components in a report. To do this, select a value to your liking (All or first 50 to name a few) from the Max Records combo box

Max Records: All on the Analysis toolbar. You can also directly input a positive integer here and press **Enter** to retrieve the corresponding records. Alternatively, you can click **Menu** > **Report** > **Max Records** to show the Max Records dialog, and then achieve the same goal. If you are making a lot of changes to the report, it may be faster to limit the number of records to 1 page while you make the changes then change it back to All to view the final result.

Manipulating a crosstab

- **Changing the group index in a crosstab**
The group index in a crosstab can be modified, namely, you can move a group to a higher or lower level. This operation can be performed on crosstabs containing two or more groups. To do this, you can simply drag a group object (row/column header) to the required destination till a blue line appears. You can also drag a column header to a row level and vice versa.
- **Rotating a crosstab**
Columns and rows in a crosstab can be exchanged. This operation is called rotating a crosstab.

To rotate a crosstab, first select it, and then do one of the following:

- Click **Menu** > **Report** > **Rotate Crosstab**.
- Click the **Rotate** button on the Analysis toolbar.
- Right-click the icon of the crosstab and select **Rotate** from the shortcut menu.

- **Expanding/Collapsing a crosstab**
For a crosstab, if it has more than one row/column group level, you can specify whether or not to enable the crosstab to be expanded in Page Report Studio, and set the default expanding/collapsing state of groups in outer levels.
- **Adjusting the width of crosstab fields according to the contents**
When the contents in the field of a crosstab need more space to completely display, you can adjust the width of the field according to its contents. To achieve it, first select the field, then right-click on it and select **Autofit** from the shortcut menu.
- **Inserting aggregate fields by dragging**
You can drag and drop fields (detail fields, aggregation fields, and dynamic formulas/aggregations) from the Resource View panel to a crosstab to add them as aggregate fields to summarize data in the crosstab.
  1. Click **Menu** > **View** > **Resource View** to open the Resource View panel if it is not displayed.
  2. Drag the required field and move the cursor above or under the desired subtotal or grand total cell, then release the cursor.
     - If the selected field has been predefined with an aggregate function,
       - When you have specified any labels to label the columns/rows/summaries in the crosstab while creating it with the report wizard, the Insert Aggregation dialog appears. You can specify a label to label the newly created summaries if necessary. Click **OK** and the field is inserted to the specified position.
       - When the crosstab has no column/row/summary labels, the field is inserted into the specified position directly.
     - If the selected field has no predefined aggregate function, the Insert Aggregation dialog appears. From the Aggregate Function drop-down list, select the function to summarize the field, then input a label to label the newly created summaries if necessary. Click **OK** and the field is inserted to the specified position. However, whether the label you specify would be generated depends on if you have added any labels to label the columns/rows/summaries in the crosstab while creating it with the report wizard. If no label is defined in the report wizard, the label you add in the Insert Aggregation dialog will be ignored.

After an aggregate field is added in a crosstab by the drag-and-drop way, if the field from which the aggregate field is created has no predefined aggregate function, you can switch the function the aggregate field uses at anytime. To do this, right-click it and select **Switch Aggregate Function** from the shortcut menu, then choose a new function from the submenu.

Manipulating a table

- **Adjusting order of columns in a table**
The order of columns in a table can be easily adjusted. To do this, drag a column header to the left or right boundary of another column header, when a blue line appears along the column boundary, release the mouse button, and you will see the order change.

The above description is for a vertical table. With regard to a horizontal table, you can do the same actions on its row headers.

- **Adjusting grouping order in a table**
A table may contain several group levels. The order of the group levels can also be adjusted. To do this, drag a group field value to the required position until a blue line appears.
- **Hiding/Deleting table columns**
A table column (for a horizontal table, the "column" corresponds to a field row) can be hidden or removed. To do this, select the cell of the column in the table header and right-click, then select **Hide Column** or **Remove Column** from the shortcut menu, and the column will be hidden or removed from the table.
- **Showing table columns**
You can specify which columns will be shown in a table. To do this, right-click the icon of the table, then on the shortcut menu, check the names of the columns you want to show from the Show Column submenu.
- **Adjusting the width of table columns according to contents**
When the contents in cells of a table column need more space to completely display, you can adjust the width of the table column according to the contents. To do this, right-click the cell of the column in the table header, then select **Autofit** from the shortcut menu.
- **Changing group direction**
You can make the group headers that are placed horizontally in a table to be displayed vertically. To do this, right-click the group header row and select **Vertical to Detail** from the shortcut menu. In addition, if the first column of a table is group column, you can specify to place the group column horizontally in a table. To do this, right-click the cell of the group column in the table header, and select **Horizontal to Detail** from the shortcut menu.
- **Rotating a table**
You can rotate a table to switch its appearance between the horizontal and vertical layout modes by doing one of the following:

- Click **Menu**> **Report** > **Rotate Table**.
- Click the **Rotate** button on the Analysis toolbar.
- Right-click the icon of the table and select **Rotate** from the shortcut menu.

- **Inserting table columns**
You can insert a new column in a table and it could be a common column, detail column, summary column, or group column.

- **To insert a common column into a table:**
  1. Right-click any cell in the table header, or right-click the icon of the table.
  2. On the shortcut menu, click **Insert** > **Common Column**.
- **To insert a detail or summary column into a table:**
  1. Right-click any cell in the table header, or right-click the icon of the table, then select **Insert** > **Detail Column/Summary Column** from the shortcut menu.
  2. In the corresponding insert column dialog, specify the resource you want to use for the new column, then click **OK**.
- **To insert a group column into a table:**
  1. Right-click any cell in the table header, or right-click the icon of the table, then select **Insert** > **Group Column** from the shortcut menu.
  2. In the Insert Group Column dialog, select the group object you want to use for the new group column from the Resources box and click to add it as the group by field, then specify the sorting direction of the group in the Sort column.
  3. Specify the positions of the group by fields: Group Above, Group Left Above, or Group Left.
  4. Repeat the above steps to add more groups if required.
  5. Click **OK** to insert the columns.

     The next time when you open the Insert Group Column dialog to add more group columns, all the added group by fields will be listed in the dialog. You can choose to remove or edit them if required.

     Note: If you right-click any cell in the table header and use its shortcut menu to insert a common, detail or summary column, the Insert Column dialog will appear for you to specify the position of the column to be inserted, which can be before or after the column in which the cell you click on is, however, if you use the table shortcut menu to insert the column,
     - If it is a common column, the column will be inserted as the last column in the table.
     - If it is a detail/summary column, the column will be inserted after the last detail/summary column, or as the last column in the table when there is no detail/summary column.
- **Converting table columns**
You can convert a group column into a detail column, and vice versa.

- To convert a group column into a detail column, select the cell of the group column in the table header, right-click and select **Convert to Detail** from the shortcut menu, then the conversion is done.

- **To convert a detail column into a group column:**
  1. Select the cell of the detail column you want to convert in the table header, right-click and select **Convert to Group** from the shortcut menu.
  2. In the Select Group Position dialog, specify the position for the newly converted group by field.
  3. Click **OK** to save the changes.
- **Aggregating on a detail column**
You can summarize the data in a detail column if required. To do this:
  1. Select the cell of the detail column in the table header, right-click it and select **Aggregate On** from the shortcut menu.
  2. In the Aggregate On dialog, specify a function from the Function drop-down list to summarize the data.
  3. When done, click **OK**.
     - If the table has groups, you will find data in each group level and the whole table are summarized respectively in the column.
     - If the table has no groups, the summary will be based on the whole table.

When you finish summarizing a detail column, you will find a dynamic aggregation object is created at the same time which is given a default name Function_DetailFieldName in the Dynamic Resource > Aggregations list in the Resources View panel and you can use it again in the current report if required.

Note: If a table is created in JReport Designer, you can add, convert columns in the table, or aggregate on its detail columns in Page Report Studio only when data fields used by the table can be converted to corresponding view elements. See the note in Analytic reporting for details.

Manipulating a banded object

- **Hiding/Showing a panel in a banded object**
A panel in a banded object can be hidden or shown. To do this, right-click the icon of the banded object, then on the shortcut menu, click the item which indicates the panel name from the Show submenu. For a panel which is shown, the item is with a check mark, and vice versa. This operation is also applicable for hiding/showing a row in a table.
- **Hiding/Showing DBfields and labels in a banded object**
The DBFields and their corresponding labels in a banded object can be hidden or shown. To do this, right-click the icon of the banded object, then on the shortcut menu, click the fields and labels you want to show from the Show Field submenu. For a field or label that is shown, it will be marked with a check mark, and vise versa.
- **Expanding/Collapsing a group panel in a banded object**
Group panels in a banded object can also be expanded or collapsed.

Manipulating a chart

- **Modifying the definition of a chart**
You can modify the definition of a chart, including the chart type, data display, and style. To do this:
  1. Right-click the icon of the chart or any part of the chart other than the legend and label to show a shortcut menu, and then select **Format Chart** from the shortcut menu to display the Chart Definition dialog.

     Note: In the event that the chart is built in JReport Designer, the Format Chart command will be available only when Page Report Studio can convert data fields used by the chart to corresponding view elements. See the note in Analytic reporting for details.
  - In the Chart Type tab of the Chart Definition dialog, specify the type for the chart.
  - In the Display tab, change the group and aggregation object used by the chart.
  - In the Style tab, modify the style for the chart as required. If there is only one style available, this style will be applied to the chart by default and the Style tab will be hidden from the dialog.
  - Upon finishing, click **OK** to apply the modifications.

    For details about how to modify the chart definition with the Chart Definition dialog, see Creating a chart report.

    In addition, if you only want to change the chart type, no matter whether the chart is originally created in JReport Designer or Page Report Studio, you can achieve it by doing one of the following:
    - Right-click the chart and on the shortcut menu, select the required type from the Chart Type submenu, which lists all the chart types and subtypes (the current one and the inapplicable subtypes are grayed out).
    - Select the chart, click the **Chart Type** button on the Analysis toolbar, and then select a suitable subtype from the drop-down menu.
- **Formatting chart elements**
The elements (platform, paper, legend, axes and labels) in a chart can be formatted to suit your requirement.

- To format the platform/paper/legend of a chart, right-click on the chart and select **Format Platform/Format Paper/Format Legend** from the shortcut menu. In the displayed dialog, specify the settings as required. For details about the format options, refer to Format Platform dialog, Format Paper dialog and Format Legend dialog.

- To format an axis, right-click on the chart and select the axis from the Format Axis submenu. In the displayed dialog, specify the axis settings as required. For details about the format options, refer to Format Category(X) Axis dialog, Format Value(Y) Axis dialog and Format Value(Y2) Axis dialog.

- To format a label, right-click the label and select **Format Label** from the shortcut menu. In the Format Label dialog, set the properties according to your requirement.

Manipulating geographic map group markers

- **Going up/down on geographic map group markers**

- For the group level that is higher than some other group levels in a geographic map component, point to its group marker, right-click it and select **Go Down** from the shortcut menu to jump one group level down.
- For the group level that is lower than some other group levels in a geographic map component, point to its group marker, right-click it and select **Go Up** from the shortcut menu to jump one group level up.

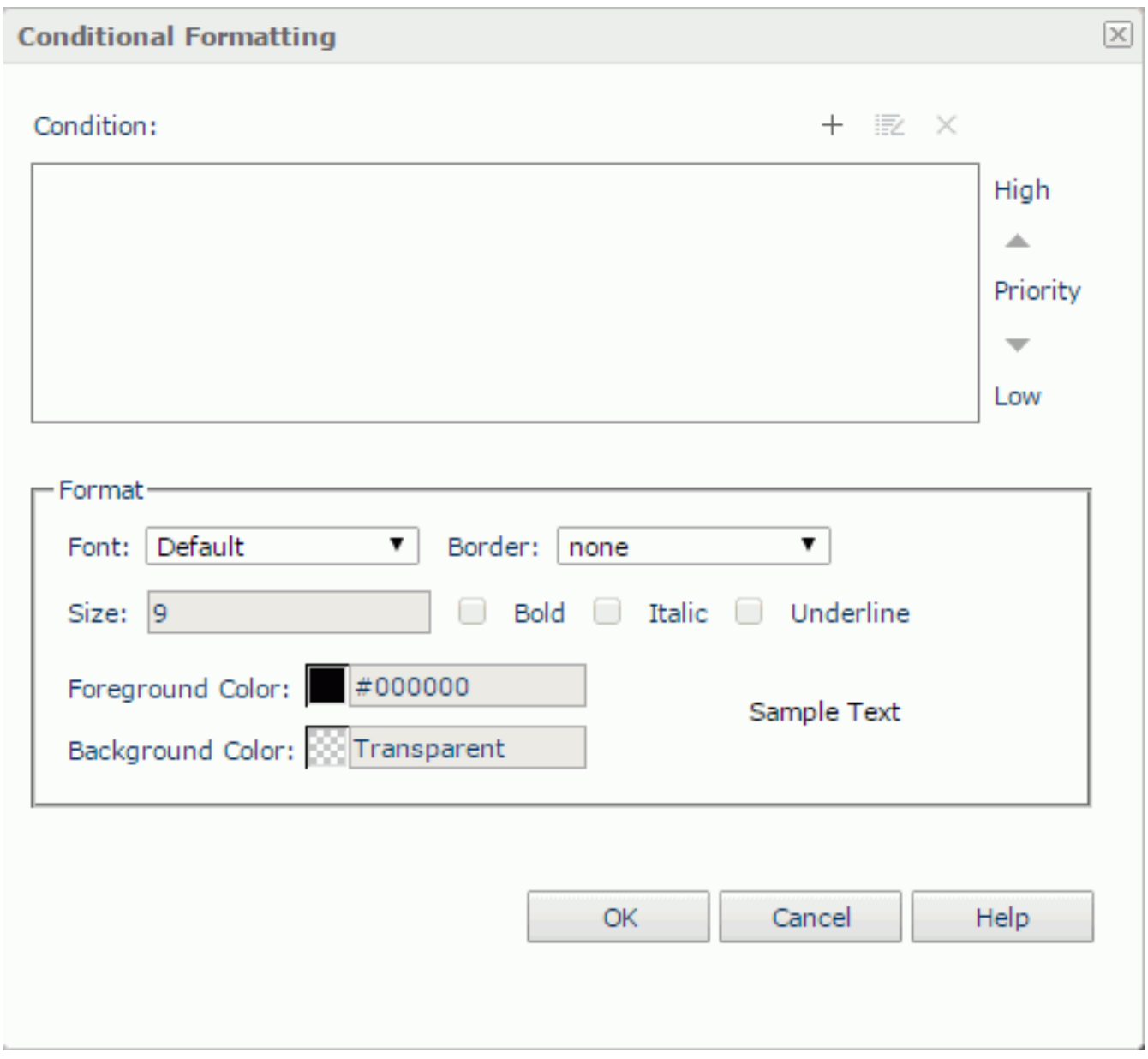
- **Showing/Hiding geographic map group markers**
By default, all visible group markers are shown. To hide them, right-click the geographic map (not the group markers) and select Hide Markers from the shortcut menu. If you want to show them again, right-click the geographic map and select Show Markers from the shortcut menu.

Adding conditional formats to fields

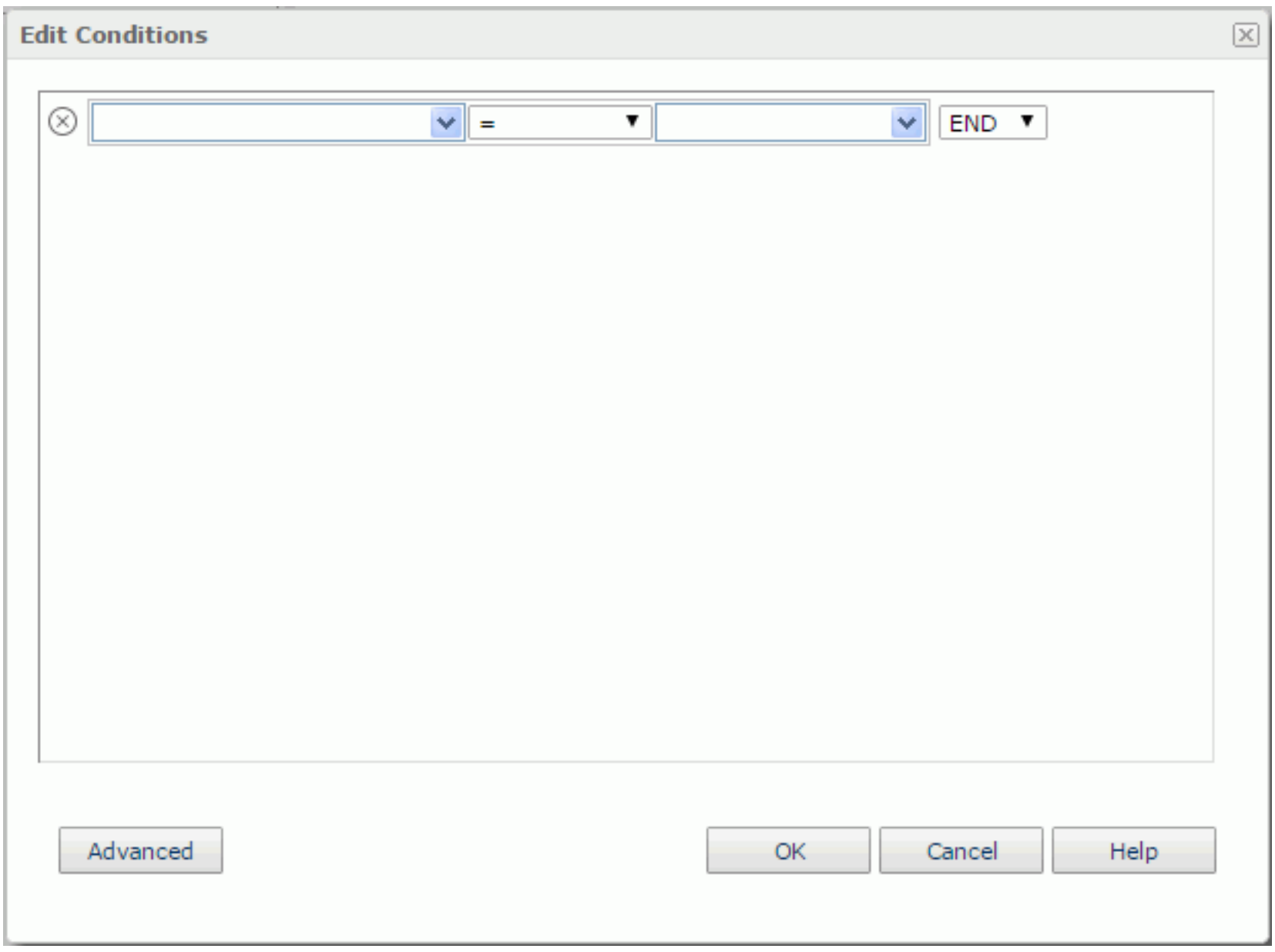
You can add some conditional formats to a field, which refer to the DBField, parameter field, formula field, summary field, and the special field Page Number or User Name, then when the specified condition is fulfilled, the defined format will be applied to the field values for highlighting.

To add conditional formats to a field,

1. Right-click the field and select **Conditional Formatting** from the shortcut menu. The Conditional Formatting dialog appears.



2. Click the button + to open the Edit Conditions dialog to define the condition as required.



There are the basic and advanced modes of the dialog for you to define either simple or complex condition expressions. See Applying filters to business views for details about how to define a condition.

3. When done, click **OK** to save the condition.

   The newly added condition will then be displayed and highlighted in the Condition box in the Conditional Formatting dialog.

4. In the Format box, set the format which will be applied to values of the field when the specified condition is fulfilled, for example, the font face, font size, font color, and so on.

5. Repeat the above steps to add more conditions and define the format for each condition as required.

   To edit a condition, select the condition in the Condition box, then click . In the Edit Conditions dialog, edit the condition expressions as required.

   To remove a condition and the corresponding format, select the condition in the Condition box and click .

   To adjust the priority of a condition, select the condition in the Condition box and then click or .

6. Click **OK** to apply the conditional formats to the field.

Converting between components

Converting between components enables you to view and analyze data from different aspects with different focuses. In Page Report Studio, you can convert a crosstab into a chart and vice versa, however, if a crosstab/chart is designed in JReport Designer, to enable the conversion, you need to make sure that data objects used by the crosstab/chart can be converted to corresponding view elements (see the note in Analytic reporting for details).

Note: A JReport Live license for JReport Server is required in order to use this feature. If you do not have the license, contact your Jinfonet Software account manager to obtain it.

Converting a crosstab into a chart

To convert a crosstab into a chart:

1. Click anywhere in the crosstab, when the icon appears at its upper left corner, click the icon to select the crosstab, then do any of following:
   - Right-click the icon and select **To Chart** from the shortcut menu.
   - Click **Menu** > **Report** > **To Chart**.
2. The Convert Data Fields dialog may appear for your confirmation on converting data fields of the crosstab to view elements of a business view. Click **OK** to confirm and the To Chart dialog appears.
3. In the Chart Type tab, specify a suitable type for the chart. With a certain type specified, you can further define the chart as a combo chart by clicking **<Add Combo Type>** in the Chart Type Groups box.
4. In the Display tab, the Resources box lists all the view elements used in the selected crosstab including group and aggregation objects. The chart can only be defined based on the view elements listed. Add a group object from the Resources box to the Category box, and so to the Series box, and aggregation objects to the Show Values box respectively.
5. In the Style tab, set the style for the chart as required.

   If the crosstab is in a banded object, by default, the chart converted from the crosstab will take on the style of the banded object. If you want to apply another style to the chart, uncheck the **Inherit Style** option and choose the desired style in the Style box. However, when there is only one style available, this style will be applied to the chart by default and the Style tab will be hidden from the dialog.
6. Click the **OK** button to finish the conversion.

Converting a chart into a crosstab

To convert a chart into a crosstab:

1. Click anywhere in the chart, when the icon appears at its upper left corner, click the icon to select the chart, then do any of following:
   - Right-click the icon or any part of the chart except for the legend and label, then click **To Crosstab** on the shortcut menu.
   - Click **Menu** > **Report** > **To Crosstab**.
2. The Convert Data Fields dialog may appear for your confirmation on converting data fields of the chart to view elements of a business view. Click **OK** to confirm and the To Crosstab dialog appears.
3. In the Display tab, select a group object in the Resources box and click to add it as a group field to the Columns or Rows box; select an aggregation object and click to add it as an aggregate field to the Summaries box. Repeat these to add more aggregate fields.

   In the Display Name column, you can edit the display name of a group field or aggregate field, and the Sort columns allow you to specify a sorting manner on a group field.

   If you want to remove any group/aggregate field, select it and click .

   To adjust the order of group/aggregate fields, select a group/aggregate field and click or .
4. In the Style tab, apply a style to the crosstab as required.

   If the chart is in a table or banded object, by default, the crosstab converted from the chart will take on the style of the table or banded object. If you want to apply another style to the crosstab, uncheck the **Inherit Style** option and choose the desired style in the Style box. However, when there is only one style available, this style will be applied to the crosstab by default and the Style tab will be hidden from the dialog.
5. Click **OK** to finish the conversion.

Note: Additional values are supported only in chart. If you convert a chart with additional values into crosstab, the additional values are not converted together with the chart.

Sorting report data

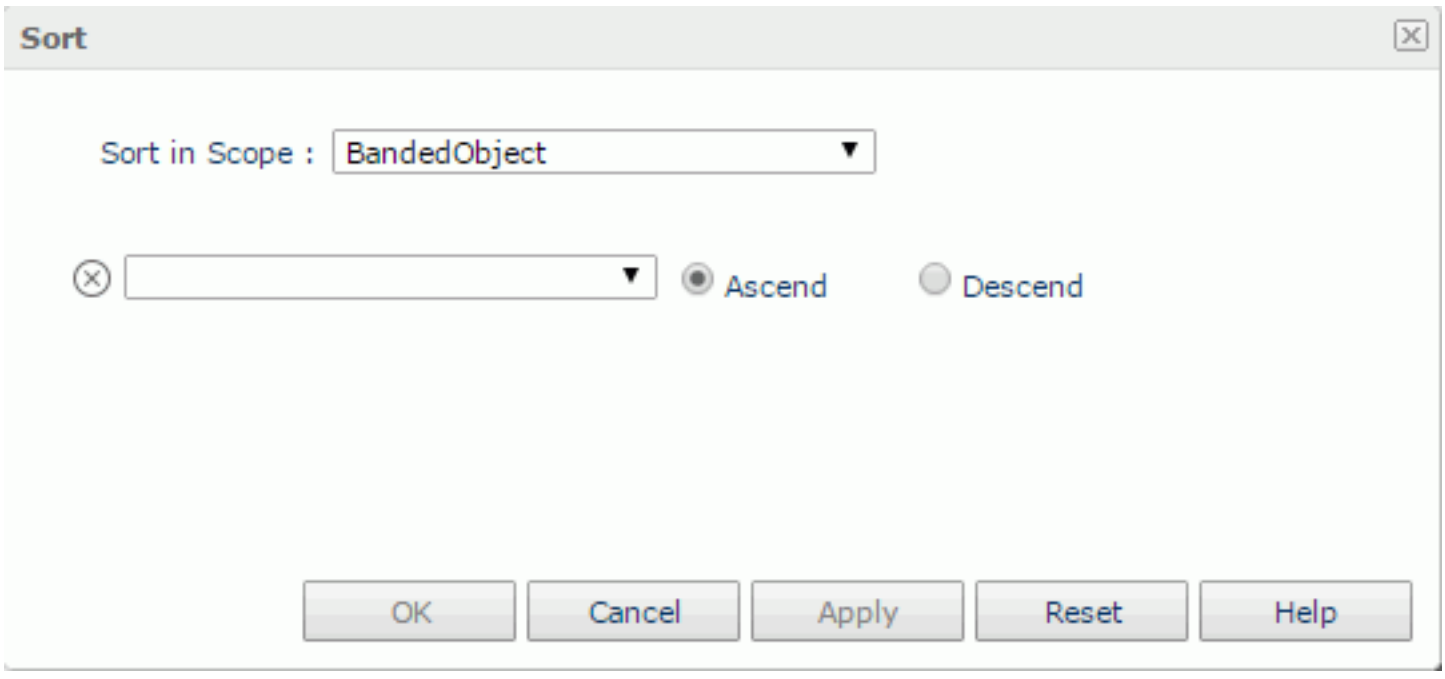
You can sort the records in a banded object or table, and the groups in a certain group level of the banded object or table if you have defined one or more group levels. If you want the data of other types of view elements to be sorted, you should put the view element into a banded object or table and make the data of the view element inherit from the banded object or table.

- **Sorting records**: Changing the order of records in the whole banded object or table, or in each group if there exists one or more group levels. The sorting scope is the whole banded object or table.
- **Sorting groups at a group level**: Changing the order of groups at the specified group level, that is, the groups will be sorted by value of the first record in each group on the related field. The sorting scope is the group level.

You can achieve the above by using the Sort dialog, shortcut menu, or labels.

Using the Sort dialog

1. Click **Menu** > **Report** > **Sort**, or the **Sort** button on the Analysis toolbar. The Sort dialog appears.



2. From the Sort in Scope drop-down list, select a banded object/table or a group field on which the sort condition will be based.
3. From the field drop-down list, select the field on which to sort the data, then set the sort order to Ascend or Descend.
4. If you select a banded object/table in Step 2, you can click + to add a new row of sorting condition if required. Click ⇧ or ⇩ to move a row up or down so as to set the sorting priority, ⊗ to delete the corresponding sorting condition if it is unwanted.

   If what you select in the Sort in Scope drop-down list is a group field, then only one sort condition can be composed.

   To retrieve the opening status of this dialog, click the **Reset** button.
5. Click **OK** to accept the settings and to reload the result.

Using the shortcut menu

1. Point to any value of a detail field or group field by which to sort the data in the banded object/table, and then right-click.
2. Choose the command **Sort** > **Ascend** or **Sort** > **Descend** from the shortcut menu.

   If you right-click on a detail field value in step 1, the sorting will affect the order of detail records in the banded object or table; if it is a group field value, the order of groups in the group level represented by the group field will be rearranged.

To remove the sort condition on a field, click **Sort** > **No Sort**.

Using labels

You can use a label to control the sorting order for a certain field. This feature needs to be enabled at report design time.

1. In JReport Designer, select a label in a banded object/table in a page report, and then set the Sortable property of the label to true.
2. Set the field by which you want to sort records as the value of the label's Bind Column property.
3. Save the report and publish it to JReport Server.
4. Run the report in Page Report Studio, and you can click the sort button beside the label to switch the sort direction between Ascend and Descend.

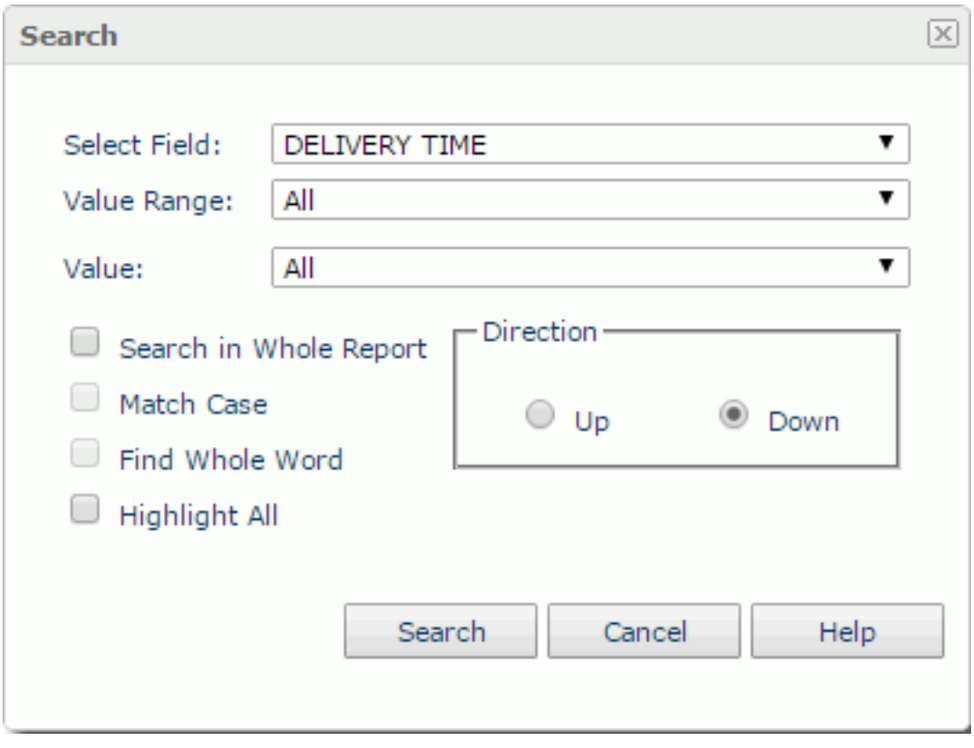
Notes:

- You cannot sort the data by a global type formula.
- You can also perform sorting by using the shortcut menu for a label in the same way as for a field value, provided you have set its Bind Column property.
- When sorting data using either shortcut menu or label, you may notice that the corresponding sort expression displays in the Sort dialog if you open this dialog.
- If you use the shortcut menu to sort the report data by a field and then sort by another field, the later sort condition will replace the former one.
- JReport allows you to define display names for fields to be showed in the Sort dialog.
- You can right-click an object in a banded object and select the Reset item from the shortcut menu to reproduce the data of the banded object using the data cached in the data buffer. This will clear all sort and filter conditions except for those predefined in JReport Designer.
- Administrators can customize the buffer size for sorting of each report in the Configuration page of JReport Server so as to improve performance.

Searching for text in a report

You can use the Search dialog to find text in the values of a certain field or in the whole report content.

To show this dialog, click **Menu > Edit** > **Search**, click the **Search** button on the Standard toolbar, or right-click a field value or label (or object such as text box) and click **Search** on the shortcut menu.



- **To find text in the values of a particular field:**
  1. Make sure the Search in Whole Report option in the Search dialog is NOT checked.
  2. Select the field from the Select Field drop-down list.
  3. Set the range with which to search for the value from the Value Range drop-down list.
  4. Select the field value you want to search for from the Value drop-down list.

     Note: If All is selected in the Value Range drop-down list, the only item in the Value drop-down list will be All and you cannot change the value, in which case, when you submit the search, JReport will search for all the values of the selected field.
- Specify whether or not to match case, whether or not to match whole word, whether or not to highlight all the matching values and the searching direction.
- Click the **Search** button.
- **To find text in the report content:**
  1. In the Search dialog, check the **Search in Whole Report** checkbox.
  2. Type the string you want to search for in the Value box.
  3. Set the other options such as the searching direction.
  4. Click the **Search** button.

Notes:

- Finding text in the values of a particular field is not supported on crosstabs and charts.
- If you check Highlight All in the Search dialog, to clear the highlighting in the search result, uncheck the option and submit the search again, or refresh the report.
- If you have not selected the Search in Whole Report option, you will not be able to search special fields for strings.
- JReport allows you to define display names for fields to be shown in the Search dialog.

Working with reports via URL

Besides working on interfaces, most of the Page Report Studio operations such as filter, sort, search, save and so on, can also be accomplished via URL. When using URLs to access Page Report Studio, you should follow the specifications described in this section.

The URL in Page Report Studio is:

`http://localhost:8888/dhtml?sessionid=XXXXXXX&rptsetid=XXXXXXX&rptname=XXXXXXX&op=...`

- **http**
  The web service protocol.
- **localhost**
  The host address.
- **8888**
  The port.
- **dhtml?sessionid=XXXXXXX&rptsetid=XXXXXXX&rptname=XXXXXXX&**
  The path.
- **?**
  URL separator.
- **=**
  URL separator.
- **&**
  URL separator.
- **op**
  The key of operation in Page Report Studio.

There are some methods in the file API.js located in `<intall_root>\public_html\javascript\dhtml`. These methods are the tools you can use to generate URLs. The result of each of these functions will be a URL.

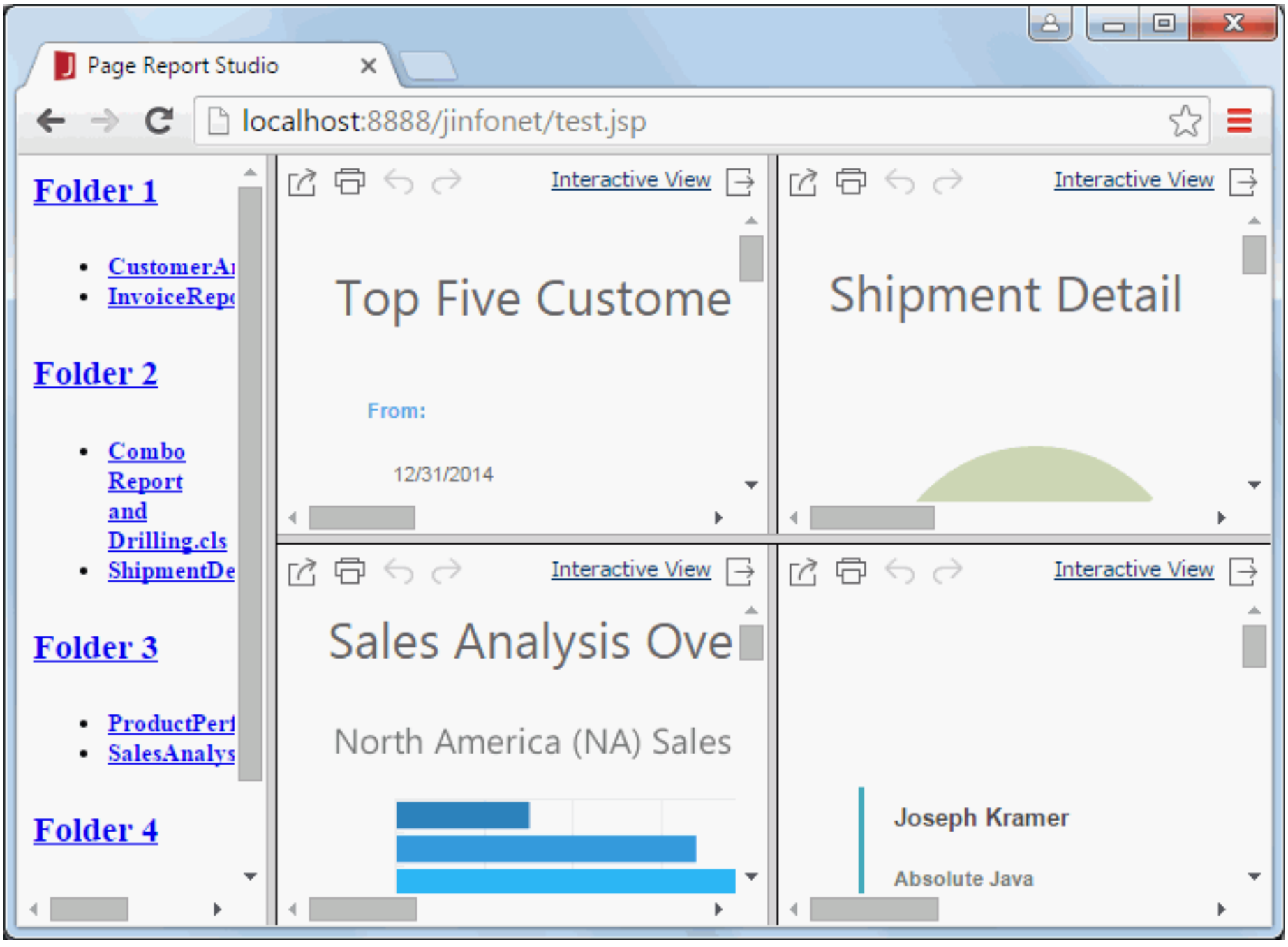
Parameter Description	Key	Value
Download Report		
Operation code	op	51
Export report type	ty	2 for PDF, 3 for PostScript, 4 for RTF, 5 for Text, 6 for Excel, and 7 for XML.
Example	op=51&ty=2	
Note	• The order begins with 2. • A file of HTML type cannot be downloaded.	
Prototype	function user_downloadReport(type)	
Exit		
Prototype	function user_exit (popwin)	
Navigate Page		
Operation code	op	24
Page number	pn	1...n
Example	op=24&pn=2	
Note	The page number begins with 1 and is less than total page number.	
Prototype	function user_firstPage() function user_lastPage() function user_nextPage() function user_prevPage()	
One Step Filter		
Operation code	op	25
The column name	col	Mapping the name of a column in JReport.
Operator of filter condition	operator	eq (equal to), gt (greater than or equal to), geq (greater than), lt (less than), leq (less than or equal to), neq (unequal to)
The column value	value	
Logic of filter condition	logic	AND, OR, END
Instance name of the column	comp	component instance name
Example	op=25&col=CusID&operator=eq&value=1&logic=OR&col=CusName&operator=0&value=ZhangKe&logic=2&comp=convert_sectionObject	
Note	The value is case-sensitive.	
Prototype	function user_oneStepfilter(columns, operators, values, logics, instanceName)	
One Step Search		
Operation code	op	128
The column name	column	Mapping the name of a column in JReport.
The column value	value	the value to search for
Whether or not to search content only	isContent	true/false
Whether or not to match case	isMatchCase	true/false
The search direction	isUp	true/false
Whether or not to match whole word	isWholeWord	true/false
Example	op=128&column=CusID&value=1&isContent=true&isMatchCase=false&isUp=false&isWholeWord=false	
Prototype	function user_oneStepSearch(value, isContent, colName, isUp, matchcase, isWholeWord)	
One Step Sort		
Operation code	op	12
The column name	col	Mapping the name of a column in JReport.
The column sort order	ord	true/false
Instance name of the column	comp	component instance name
Example	op=12&col=CusID&ord=true&col=CusName&ord=false&comp=convert_sectionObject	
Note	The value is case-sensitive.	
Prototype	function user_oneStepSort(columns, sorts, instanceName)	
Open Report		
JSP page	/dhtmljsp/dhtml.jsp	
Action	jrs.cmd	jrs.try_vw
Report name	jrs.report	path/name
Catalog path	jrs.catalog	
Result type	jrs.result_type	int
Example	http://localhost:8888/dhtmljsp/dhtml.jsp?jrs.report=%2fSampleReports%2fEmployee Information.cls=&jrs.cmdjrs.try_vw&jrs.catalog=%2fSampleReports%2fSampleReports.cat	
Note	• The path delimiter is "/". If you encounter problems, you can replace it with "%2f". • The version number can be none, as with the example.	
Open RSD File		
JSP page	/dhtmljsp/dhtml.jsp	
Result path	jrs.path	path/name
Result name	jrs.result	path/name
Result version number	jrs.rst_version	int
Example	http://localhost:8888/dhtmljsp/dhtml.jsp?jrs.resource_path=/USERFOLDERPATH/admin/test&jrs.file=1980996366.rsd	
Note	• The path delimiter is "/". If you encounter problems, you can replace it with "%2f". • The version number cannot be none.	
Redo		
Operation code	op	85
Example	op=85	
Prototype	function user_redo()	
Refresh		
Operation code	op	83
Example	op=83	
Prototype	function user_refresh()	
Reset		
Operation code	op	76
Example	op=76	
Prototype	function user_reset()	
Save Report		
Operation code	op	82
Example	op=82	
Prototype	function user_saveRpt()	
Search Next		
Operation code	op	34
Example	op=34	
Show Export to Dialog		
Prototype	function user_showSaveResultDialog()	
Show Help		
Dialog ID	HELP_OP	182
Example	http://localhost:8888/dhtmljsp/help.jsp?sessionid=XXXXXXX&rptsetid=XXXXXXX&rptname=XXXXXXX&HELP_OP=XXX	
Note	The URL gets the body of a window, you should put it as xxx in window.open("xxx", ..)	
Prototype	function user_showHelp(helpId)	
Show Panel or Dialog		
Prototype	function user_showUserPanel() function user_showTOC() function user_showDHTMLView() function user_showToolbox() function user_showSortDialog() function user_showFilterDialog() function user_showNewRptDialog() function user_showOpenRptDialog() function user_showSaveAsDialog() function user_showPageSetupDialog() function user_showSaveResultDialog() function user_showPageSetupDialog() function user_showPrintDialog()	
Undo		
Operation code	op	84
Example	op=84	
Prototype	function user_undo()	
Zoom		
Operation code	op	50
Proportion	val	0~400
Example	op=50&val=250	
Note	val must be an integer between 0 and 400	
Prototype	function user_zoom(value)	

Opening multiple reports in one session

You can open multiple reports in Page Report Studio in one session. This means that each time you run a page report, it will open in a new window. Also, when working in the embedded mode, you can assign frames to the reports, so that more than one page report can be viewed in one window at the same time.

In order to illustrate how to view multiple frames in one window, a demo has been provided for you. Follow the steps below:

1. Go to `<install_root>\help\samples\JSPSamples\OpenMultipleReports`.
2. Copy the two JSP files in the folder to `<intall_root>\public_html\jinfonet`.
3. Access JReport Server using `http://localhost:8888/jinfonet/test.jsp`.
4. From the left panel, browse to the report you want to open and click its name. It will then be displayed in the corresponding frame.



Tuning Page Report Studio performance

JReport provides you with methods for adjusting Page Report Studio performance. You can limit the number of page reports open simultaneously in Page Report Studio by setting the Page Report Studio preferences on the JReport Administration page. Furthermore, you can modify a property file to control the Action Task Manager, which can improve Page Report Studio service performance.

Limiting the number of simultaneously open reports

Whether or not an open page report interacts with the server, it holds many resources. As a result, opening many reports will decrease server performance. Thus, JReport allows the administrators to specify the maximum number of page reports that can be open at the same time in Page Report Studio so as to prevent large numbers of page reports from being open simultaneously and to improve the server performance.

To set the maximum number of page reports that can be open at the same time:

1. On the JReport Administration page, click **Profile** on the system toolbar and then select **Customize Profile** from the drop-down menu.
2. Click the **Page Report Studio** > **Properties** > **Advanced** tab.
3. Check the **Maximum Number of Open Reports** option and type the number to your requirement in the text box.
4. Click **OK** upon finishing.

If the number of open reports exceeds the limit, an error page will be displayed prompting you to close one before opening a new one.

Note: The Maximum Number of Open Reports option works together with the maximum number of concurrent reports allowed by your product license. Of these two values, whichever is smaller will be used as the real maximum number of open reports allowed. For example, if the value of the Maximum Number of Open Reports option is 10, and the number that the license allows is 20, 10 will be used as the maximum number of page reports that can be opened simultaneously.

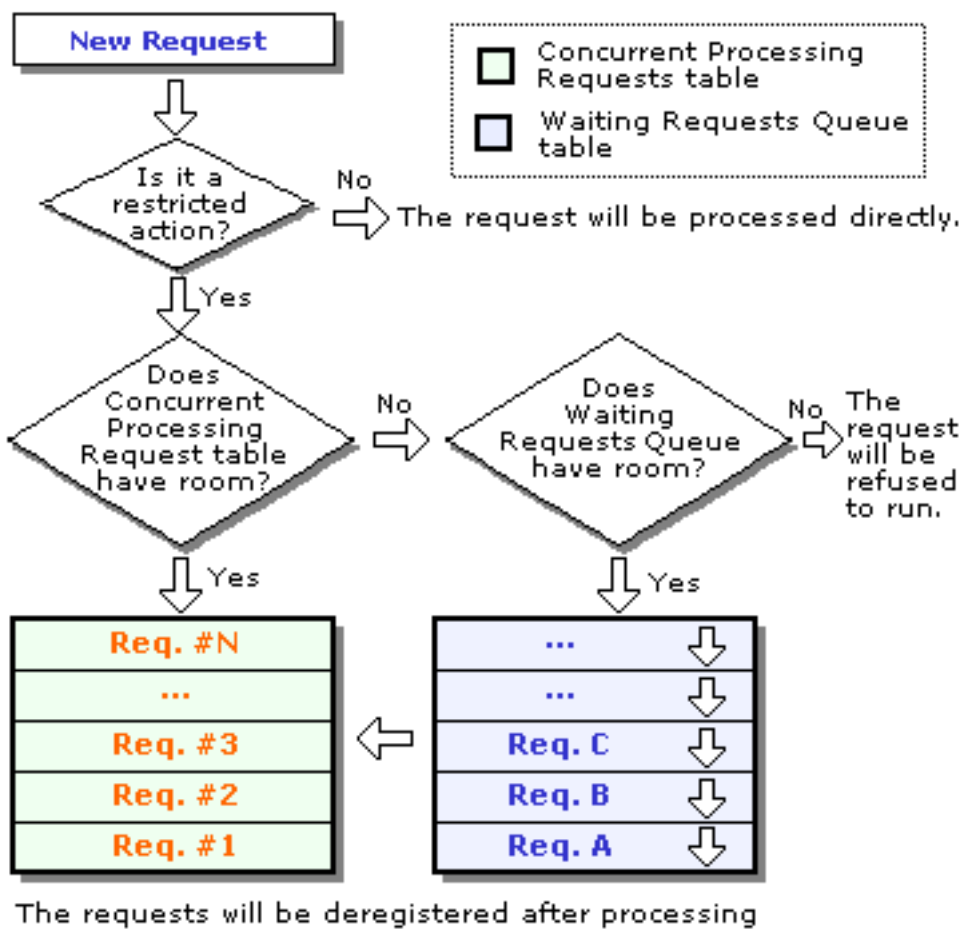
Action Task Manager

Some Page Report Studio operations require a large amount of memory and CPU processing power. The Action Task Manager improves Page Report Studio service performance by preventing a large number of actions from being run simultaneously.

The Action Task Manager coordinates Page Report Studio actions through two fixed-size tables:

- **Concurrent Processing Table** - This registers the requests that are currently being processed by the Page Report Studio service.
- **Waiting Requests Queue** - This registers the requests that are waiting for being processed by the Page Report Studio service.

Note: Only certain operations that consume considerable hardware resources need to be prevented from being run at the same time. You can define which kind of requests need to be queued before being processed.



When a new Page Report Studio request reaches the server, it will be processed according to the following flow:

1. The Page Report Studio service determines whether the requested operation is a restricted action. If it is, the Action Task Manager will take over the request. Otherwise, it will be processed directly, without being managed by the Action Task Manager.
2. If the Concurrent Processing Table is full, the restricted request will be assigned to the Waiting Requests Queue. If the queue is full, the Page Report Studio service will refuse the request and return a warning message.
3. After the request has been processed, it will be de-registered from the Concurrent Processing table. The Page Report Studio service will then automatically continue to process the requests in the Waiting Requests Queue.

By using the property file dhtml.properties provided in Page Report Studio, you can balance the server load by adjusting table sizes and specify which kind of requests are managed by the Action Task Manager.

dhtml.properties

The dhtml.properties file is located at `<install_root>\bin`. It allows you to control three major options for the Action Task Manager:

- **Specifying the size of the Concurrent Processing Table**

  Use `queue.actions.max.concurrent=[integer]` to set the maximum number of requests that can be processed simultaneously. The value of this property can be equal to or larger than 0. Use 0 (default) to disable the request queue feature.

- **Specifying the size of the Waiting Requests Queue table**

  Use `queue.actions.max.pending=[integer]` to set the maximum number of to-be-handled requests that the queue can contain. The value of this property can be equal to or larger than 0. 0 means no requests will be stored in the queue. A request will either be handled by the Page Report Studio service or be rejected when the maximum limit of the Concurrent Processing table has been reached.

- **Specifying the actions that can be applied for the Page Report Studio Request Queue feature**

  These are listed below:

```
queue.actions.init=false            # Action: DHTML report initialization
queue.actions.undo=true             # Action: Undo
queue.actions.redo=true             # Action: Redo
queue.actions.drill=true            # Action: Drilling
queue.actions.drillup=true          # Action: Drilling up
queue.actions.refresh=false         # Action: Refreshing
queue.actions.filter=true           # Action: Filtering
queue.actions.sort=true             # Action: Sorting
queue.actions.search=true           # Action: Searching
queue.actions.finishNewReport=true  # Action: Finishing creating a new report
```

  These properties will only work when the queue feature has been enabled by setting `queue.max.concurrent.actions>0`.

  True - The action will be handled by the Action Task Manager.
  False - The action will not be handled by the Action Task Manager, but be directly processed by the Page Report Studio service without being queued.

Web Report Studio

JReport provides the web reporting solution for faster and simpler design and creation of reports using a web browser.

Web reports are viewed using a new interactive viewer called Web Report Studio. Web Report Studio provides a much nicer end user experience with many powerful features for interfacing with a report such as changing parameters without re-running the report. In addition, Web Report Studio still supports exporting the report to all supported output formats.

The data sources that can be used to create web reports are business views that are resources built on top of queries. Business views are created and managed in JReport Designer. In JReport Server, web reports are created via the Web Report Wizard using a browser and are opened and edited via Web Report Studio.

Using Web Report Wizard, it is easy to create complex reports with multiple components in a tabular style layout. Web reports also allow for company logo and titles to be placed on the top of the page for more formal presentation.

A web report template contains only one report and uses .wls as the file suffix. Web reports are stored in the JReport Server resource system and follow the server resource and version management rules such as archive policy and permission setting.

JReport Designer supports creating, opening, and editing of web reports. JReport Designer also allows for web reports created on JReport Server to be downloaded to Designer and further edited.

This chapter covers the following topics to help you better interact with web reports:

- Why web reports and when to choose them
- Components supported in web reports
- Web Report Studio window elements
- Creating web reports via wizard
- Editing web reports in Web Report Studio

Notes:

- A JReport Live license for JReport Server is required in order to use Web Report Studio. If you do not have the license, contact your Jinfonet Software account manager to obtain it.
- Web Report Studio cannot run on Internet Explorer 8.

Why web reports and when to choose them

Web Report Studio displays web reports (which are also called web layout reports) that are aimed at easier and faster report creation and design, faster report execution, easier customization, and better presentation style using a newer Rich Internet Application (RIA) Web 2.0 interface. Web reports also support agile development techniques such as continuous integration by allowing report templates to be updated by both Web Report Studio and JReport Designer.

- **Fewer functions**
  Web reports (.wls) support a subset of functions of JReport page reports (.cls). The basic and essential functions not only guarantee a good report presentation, but also make the report design experience easier for a new user of JReport.

- **Single report solution**
  Only one report in a web report speeds up the report running process as compared to a multiple-report page report.

- **Tabular style layout**
  The creation of a page report using the Standard Report Wizard can only create one data component (table, crosstab, chart, or banded object) using the wizard. The Web Report Wizard provides a tabular style layout in which you can place a table, crosstab or chart in each tabular cell so as to achieve a holistic layout with multiple components from the very beginning.

- **Predefined report templates**
  Web reports allow you to choose a starting template so you can predefine the template to include standard features such as company logo, company name, privacy notices or any standard items and styles you want your users to start with.

- **Fast report rendering**
  At runtime, Web Report Studio provides much higher performance when viewed from a browser compared to viewing a page report using Page Report Studio. Using Page Report Studio, all of the user action requests must be sent to JReport Server which renders the new page on the server and updates the browser view. Using Web Report Studio, many of the actions which require only a change in rendering the view are done locally on the client in the browser. By using Web Report Studio, JReport Engine is structured so that as much of the processing as possible is completed on the client side allowing much higher scalability for JReport Server allowing the server to handle more simultaneous users.

- **Creation and edition in both JReport Server and Designer**
  Web reports created using the Web Report Wizard can be downloaded from JReport Server and edited in JReport Designer and web reports can be created in JReport Designer and published to server just like .cls reports. However, page reports created using the Standard Report Wizard can only be edited in the server. Designer can view these reports but not modify them and publish them.

  Also, JReport Designer can be used to create web reports which can be run in Web Report Studio on the server and saved as a template to use in Web Report Studio. JReport Designer cannot directly create the template but can create the report and then using Web Report Studio, save the report as a template (.wsld).

- **Standard banded objects not supported**
  Page reports support standard banded objects which are not supported by web reports.

Components supported in web reports

Components are the objects that you can place in a report. JReport provides a full set of components that allow you to present and control the report data and presentation in a wide variety of ways.

Web reports support the following report components:

- **Labels**
  A label is an object that contains a string. It is typically a brief description used to identify a field or other value nearby.
- **Images**
  An image is a digital representation of a picture. The following image types are supported in web reports: .gif, .jpg, and .png.
- **DBFields**
  DBFields, or database fields, are fields directly from columns in the database or other data source such as XML or Java objects.
- **Formulas**
  Formulas are calculated from DBFields, other formulas, summaries, and parameters, so they can present information which is not available directly from the database fields.
- **Summaries**
  A summary is a special kind of formula. A summary generates a count, average, sum, standard deviation or other transformation of a set of data values. A summary applies to a defined group of data. Summaries are required to provide the data values for charts as well as totals for table reports.
- **Parameters**
  A parameter in JReport is a variable whose value is determined at runtime. The runtime parameters help you dynamically control your report result such as filtering data.
- **Special fields**
  Special fields are defined by JReport and allow you to easily obtain system information and report-related data and add it to your report. All special fields are supported to insert into web reports in JReport Designer. However, due to the characteristics of Web Report Studio, only these can be rendered and edited in Web Report Studio: User Name, Modified Date, Modified Time, Fetch Date, Fetch Time, Print Date, and Print Time.
- **Web controls**
  Web controls are report components designed to be similar to the kinds of controls found on web pages. Currently, the following four web controls are supported in web reports: parameter control, parameter form control, filter control, and navigation control.
- **Multimedia objects**
  Multimedia objects include Flash, Real Media, and Windows Media objects.
- **Tabular**
  A tabular is a component designed to lay out other components. There is one and only one tabular in a web report.
- **Tables**
  A table gives you great control over how to present data, including placing fields, grouping them, and sorting them. It is composed of rows and columns, and each contains several cells. With such a structure a table is a good way to show any two-dimensional dataset.
- **Crosstabs**
  A crosstab summarizes data and presents the summaries in a compact row and column format.
- **2-D Charts**
  A chart organizes and graphically presents data in a way that makes it easy for end users to see comparisons, trends, and patterns in data. It represents the report data in a visually straightforward form. A chart is based on the chart platform. On the platform, the chart paper, the legend, and labels make up the chart. You can create a chart that contains only simple DBFields, or a complicated chart that contains DBFields, groups, summaries, and even formulas. Normally, DBFields, summaries, and formulas in a report are represented in a chart using chart data markers, and groups are used to produce category names and data series names. DBFields can also be used as category names.

Tip: In JReport, the components that can be bound with a data source are also referred to as data components. These components include tables, crosstabs and charts.

Web Report Studio window elements

Here are two ways of accessing Web Report Studio:

By creating a new web report

On the JReport Console > Resources page, click **New** > **Web Report**. Then follow the Web Report Wizard and click **Run**. Web Report Studio will then be displayed with the new report.

By running an existing web report

On the JReport Console > Resources page, browse to the target web report and choose the appropriate way to run the report in Web Report Studio.

Web Report Studio window elements

The main page of Web Report Studio consists of the user information bar, menus, toolbar, left panels and report area. The options for browsing or controlling a web report are as follows:

Menu/ Toolbar	Button	Tool Name	Description
File		New Report	Creates a new web report based on an existing business view.
		Open	Opens a web report.
		Save	Saves the changes of the current web report.
		Save As	Saves a copy of the web report or the report template in the current web report to server resources.
		Export	Exports the report result to disk or version in various formats.
		Page Setup	Configures the report page settings.
		Print	Displays the Print dialog for printing the web report result.
		Exit	Closes the current web report and exits Web Report Studio releasing all of the resources.
Edit		Undo	Undoes the last operation.
		Redo	Reverses the operation of Undo.
		Delete	Deletes the selected object.
		Wizard	Opens the report wizard for you to edit the selected table, crosstab or chart.
		Filter	Filters the report records according to the filter criteria you specify.
		To Chart	Converts a crosstab into a chart.
		To Crosstab	Converts a chart into a crosstab.
		Rotate Crosstab	Rotates a crosstab to exchange the axes on the crosstab in order to create a different view of the crosstab.
		Report Body Properties	Defines properties of the report body.
		Bind Data	Specifies to bind a data source to the web report.
		Unhide Components	Shows the hidden components you specify.
		Style	Applies a style to the selected components or the whole report.
View		Inspector	Shows or hides the inspector panel, which lists all objects in the current web report in a tree structure. You can select any object in the tree to edit its properties.
		Editing Marks	Shows or hides editing marks (dashed outlines for objects and report body). If the option is unselected, the editing mark will not be shown when a report object receives focus, and report objects cannot be moved or resized.
		Refresh	Runs the web report using previously provided parameters. The Refresh operation fetches the data again.
Format		Font	Specifies the font format of the selected text. Available only when a label or field is selected.
		Merge	Merges the selected tabular cells into one.
		Split	Splits the selected tabular cell into the specified number of rows and columns.
Language			Allows you to specify the language in which to display the report. Available only when Enable NLS is checked in the Profile > Customize Server Preferences > Advanced panel on the JReport Console page.
Help		User's Guide	Opens the Web Report Studio User's Guide.
		JReport Home Page	Connects to JReport Home Page.
		Technical Support	Accesses Jinfonet Technical Support.
		About Web Report Studio	Shows product information about Web Report Studio.
	English (United States)	Language	Specifies the language in which to display the report. Available only when Enable NLS is checked in the Profile > Customize Server Preferences > Advanced panel on the JReport Console page.
Standard Toolbar		New Report	Creates a new web report based on an existing business view.
		Open	Opens a web report.
		Save	Saves the changes of the current web report.
		Save As	Saves a copy of the web report or the report template in the current web report to server resources.
		Export	Exports the report result to disk or version in various formats.
		Page Setup	Configures the report page settings.
		Print	Prints the report result to a PDF file with the default page settings.
		Refresh	Runs the web report using previously provided parameters. The Refresh operation fetches the data again.
		Undo	Undoes the last operation.
		Redo	Reverses the operation of Undo.
		Filter	Filters the report records according to the filter criteria you specify.
		Delete	Deletes the selected object.
		Show Objects	Specifies which objects to show in the report.
Quick Format Toolbar		Font	Specifies the font format of the selected text. Available only when a label or field is selected.
		Background Color	Changes the background color of the selected text. Available only when a label or field is selected.
		Align	Makes the selected text left, center or right aligned. Available only when a label or field is selected.
		Merge	Merges the selected tabular cells into one.
		Split	Splits the selected tabular cell vertically or horizontally.
Context Toolbar for Table		Table Wizard	Opens the Table Wizard for you to edit the table.
		Show/Hide Detail	Hides or shows the detail columns you specify.
		Add/Remove Group	Specifies whether to add or remove the selected field as a group.
		Show/Hide Summary	Specifies whether to show or hide the selected summary field.
		Hide	Hides the selected column.
		Aggregate On	Creates a new summary directly based on the field bound with the table detail column.
Context Toolbar for Crosstab		Crosstab Wizard	Opens the Crosstab Wizard for you to edit the crosstab.
		Rotate Crosstab	Rotates a crosstab to exchange the axes on the crosstab in order to create a different view of the crosstab.
Context Toolbar for Chart		Chart Wizard	Opens the Chart Wizard for you to edit the chart.
		Swap Chart Groups	Specifies whether to switch data between the category and series axes, or between the category and value axes if there is no field on the series axes.
		Chart Type	Lists all available chart types for you to change the type of the chart.
		Chart Options	Lists more options for you to specify the layout of the chart.
Panel		Parameters	Lists all the parameters used by the current report. It is available when the current report uses parameters.
		Resources	Lists all the available resources. When a data component is selected, the Sort and Search buttons on the title bar of the panel will be enabled. For the usage about the two buttons, click here.
		Components	Lists all the available components that can be inserted into reports.
		Filter	Specifies the criteria to filter the data field. You can also remove or change existing filters.
		"Go To" Filter	After you perform the go-to-by-value or go-down actions, the panel is displayed showing the filter created by the actions. You can also click in a table, crosstab, or chart to show this panel for the component.
		Properties	Lists all the objects in the report in a tree structure with their properties.
Shortcut Menu		Show	Shows the selected fields.
		Apply Style	Applies a style to the selected component.
		Delete	Deletes the selected object.
		Autofit	Adjusts the width of table and crosstab fields according to the contents.
		Hide	Hides the selected object.
		Filter	Provides submenu items for filtering the data in the selected component or remove existing filters.
		Sort	Provides submenu items for sorting records on the selected field in ascending/descending order, or remove the sort.
		Edit Detail Table	Edits the detail table to define the detail fields of the summary.
		Go to Detail	Goes to the detailed information of the selected summary.
		Link/Edit Link	Links the selected object to a report, URL, e-mail or Blob data type field.
		Conditional Formatting	Adds some conditional formatting to the currently selected field.
		Go Down	Goes from a group which is non-bottom level in a predefined hierarchy to the one-level-lower group while applying the current selected value as a filter condition. See Go-down for details.
		Go Up	Goes from a group which is non-top level in a predefined hierarchy to the one-level-higher group. See Go-up for details.
		Go To	Goes to any group to show its record information.
		Go to By Value	Goes to any group with the current group value as a filter to show its record information.
		Select	Make the corresponding object selected.

Notes:

- Web Report Studio has two modes: View Mode and Edit Mode, and the toolbar and menu commands that are available in each mode vary. Administrators can specify the default mode that will be applied when a report is opened in Web Report Studio in the Profile > Customize Server Preferences > General tab, and you can switch between the two modes by selecting View Mode or Edit Mode from the mode drop-down menu on the Web Report Studio toolbar. However, you will not be able to switch the mode if the Show Link of View/Edit Mode option in the Profile > Customize Profile > Page Report Studio > Properties > Default tab is unchecked.
- The shortcut menu contents vary with the objects you right-click. The above table only lists some typical items. The following sections will guide you to use the shortcut menu for any object you may right-click.

Creating web reports via wizard

On the JReport Console > Resources page, you can directly create a new web report in a folder into which a catalog containing one or more business views has been published.

To create a web report:

1. Open the folder and select the catalog for the new web report from the Catalog drop-down list, then click **New** > **Web Report** on the task bar of the Resources page. The Web Report Wizard appears.
2. In the Page screen, choose a template for the report. The built-in templates contain information about Jinfonet. You can customize them to meet your own requirement. Use ... to load your company logo. Edit the text and set the font properties for company titles and report titles using Az. Click the **Page Setup** link to set the page properties. If you are an administrator with the privilege of publishing resources, you can also create a new template to save your customization.
3. In the Layout screen, select the required layout with which you want to create the report. Then, in the edit layout area, select a tabular cell and select the component you want to display in the cell. Click the **Align** drop-down list to set the component to the left, center or right of the cell. Repeat this to add component to the other cells.

   If required, you can split the selected cell horizontally or vertically by clicking the Horizontal Split or Vertical Split button, merge adjacent cells by selecting them and clicking Merge. You can also resize the tabular cells by dragging the cell border.
4. In the Bind Data screen, define the specified components (for details about how to define a component, refer to the specific topic in Inserting components). You can use the Back and Next buttons to switch between the components.
5. In the Style screen, apply a style to the report.
6. Click **Save** to save the report to the server resource tree. For details, see Saving the report.
7. Click **Run**. If any parameter is used in the report, the Report Parameters dialog appears for you to specify the parameter value, then the report is opened in Web Report Studio.

Web report templates

When creating web reports, you need to choose a starting template. A web report template stores information on the header and footer as a starting point for a web report. If you are an administrator with the privilege of publishing resources, you can save report templates to include standard features such as a company logo, company name, privacy notices or any standard items and styles you want your users to start with.

The saved report templates will be added into the templates directory on the server `<install_root>\templates`, and will be automatically loaded onto the Page screen of the Web Report Wizard for use.

Creating web report templates

To create a report template by customizing the built-in templates in the Web Report Wizard:

The Web Report Wizard provides sample templates for you to define your own templates based on.

1. In the Page screen of the Web Report Wizard, make use of Template1 or Template2 to customize your own report templates. With Template1, you can add your company logo and report title. With Template2, you can add more items such as company name and title and report sub title. Use ... to load your company logo. You can set the font properties for company titles and report titles using Az.
2. Click the **Save** button.
3. In the Save As dialog, choose the file type of **Web Report Template (*.wsld)**, specify a name for the template in the File Name text field, then click **Save** to save the template.

To create a report template by editing a report in Web Report Studio:

1. Open a report in Web Report Studio.
2. Edit the header and footer of the report.

   The report header and footer are composed by two tabulars. You can split or merge the tabular cells by clicking the button or on the toolbar, then add objects from the Components panel into the tabular cells and format them to customize the header and footer.
3. Click **Menu** > **File** > **Save As** or the **Save As** button on the Standard toolbar.
4. In the Save As dialog, choose the file type **Web Report Template (*.wsld)**, specify a name for the template in the File Name text field, then click **Save** to save the template.

Dynamically displaying the company logo

If the web report template used for creating a report contains the company logo information, you can make the logo display dynamically.

To display the company logo dynamically:

1. Select the company logo image in the report and then do either of the following:
   - Right-click the image and select **Properties** from the shortcut menu to display the Image Properties dialog, then click fx next to the Image Name option.
   - Click **View** > **Inspector** on the menu bar to display the Inspector panel, then click fx in the value cell of the Image Name property.
2. Do either of the following to change the image name.
   - Select a formula that returns an image source from the drop-down list. By default, there will be no formula available in the list. If you want to use a formula to control the image source, you need to first bind a data source to the web report, then create formulas that return image sources in the Resources panel. For details, refer to Using formulas to control showing or hiding components.
   - Select **<Edit Expression>** from the drop-down list to open the Formula Editor and edit an expression to control the image source.

     JReport provides a built-in parameter named JRS_P_LOGOURL, which enables you to specify the image path easily. You can enter return *@JRS_P_LOGOURL;* directly in the editing box of the Formula Editor to create the expression.
3. If you are in the Image Properties dialog, click **OK** to accept the change.

   If the specified formula or expression returns an image, the image is changed. If the specified formula or expression uses parameters, the parameters are added to the Parameters panel. You can specify the parameter values in the panel to dynamically change the image.

If you are an administrator with the privilege of publishing resources, you can then click **Menu** > **File** > **Save As** or the **Save As** button on the Standard toolbar to save the template as a new one for future use. However, if the company logo of a template is controlled by a formul, later when users use this template to create web reports, the web reports might not be created successfully, so JReport provides the option When web report cannot run correctly as its template uses formulas referencing bound data on the Profile page for you to determine what you want JReport to do in this case.

Renaming/Removing a web report template

To rename or remove a web report template, go to the templates directory `<install_root>\templates` on your server, then rename or delete the template file (.wsld). For each template file, there is an image file (.wsld.png) which is used to display in the templates box of the Page screen of the Web Report Wizard, as a representative of the template. You will need to rename or delete the image file when you rename or delete the template file.

Editing web reports in Web Report Studio

Web Report Studio is the web oriented page where you view and edit web reports. When a report is opened in Web Report Studio, by default it is in the view mode which provides only viewing-oriented functions. If you want to edit the report, click the Edit Mode link on the toolbar to enter the edit mode.

Pick a task from the following:

- General operations in reports
- Inserting components
- Making simple modifications to components
- Manipulating data components
- Binding links to objects
- Using dynamic resources
- Going through the report data
- Applying filters
- Using web controls
- Adding conditional formats to fields
- Applying parameters
- Sorting report data
- Applying CSS styles
- Saving the report
- Exporting/Printing the report result

General operations in reports

You can perform the following general operations in Web Report Studio:

- **Opening another web report**

  Click **Menu** > **File** > **Open** (or the **Open** button on the Standard toolbar) to display the Select a Report dialog, in which the web reports in the same folder as the current open report are listed. Select the web report you want to open from the default folder or from another folder, and then click **OK**.

- **Exiting Web Report Studio**
  If you want to close the current web report and release the resources, just click **Menu** > **File** > **Exit** (or the button **X** on the far right of the toolbar). Do not use the close button on the browser window as that may not release the resources used by the report.

- **Undoing / Redoing actions**
  You can undo or redo some actions. To do this, click **Menu** > **Edit** > **Undo** or **Redo** (or the **Undo** button or **Redo** button on the Standard toolbar).

- **Navigating component data via scrollbar**
  For tables, crosstabs and charts, you can use the scrollbar to navigate their data if the tabular cell can not display all data of the component.

- **Turning component pages**
  In Web Report Studio, if a table or a crosstab contains more than one page, a navigation bar specific for the component will be available right below the component. You can use the navigation bar to view the desired pages: click the corresponding button to go to the first page, the previous page, the next page, or the last page, or input a number in the text box to go to that page.

- **Showing / Hiding editing marks**
  You can use editing marks (dashed outlines of objects) for purposes such as aligning, moving and resizing. The editing marks are shown by default. To switch the status of the editing marks, click **Menu** > **View** > **Editing Marks**.

- **Asking for help**
  At any time, you can click **Menu** > **Help** > **User's Guide** to open the index page of the Web Report Studio User's Guide. Furthermore, you can click the Help button in any dialog to show the help document about the dialog. You can also use the Help menu to access Jinfonet Software website for more information.

Inserting components

You can insert components into a web report via the Components panel on the left of the Web Report Studio window.

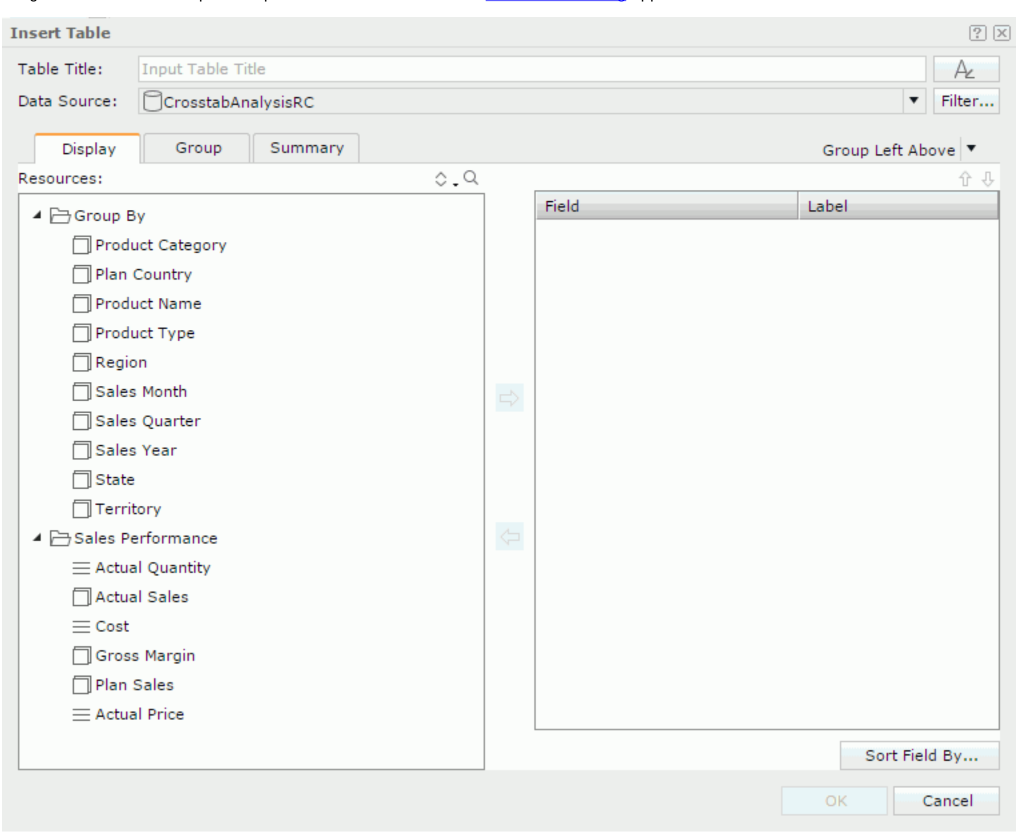
The following table lists the report areas that are valid targets for the various components.

	Report Layout Area			
Component	**Page Header / Footer**	**Report Body**	**Tabular Cell**	**Table Cell**
Chart	Y	Y	Y	N
Crosstab	Y	Y	Y	N
Table	Y	Y	Y	N
Group object	Y	Y	Y	Y
Detail object	Y	Y	Y	Y
Aggregation object	N	Y	N	Y
Formula	Y	Y	Y	Y
Label	Y	Y	Y	Y
Image	Y	Y	Y	N
Multimedia object	Y	Y	Y	N
Web control	Y	Y	Y	N

The following shows inserting a specific component in detail:

Inserting a table

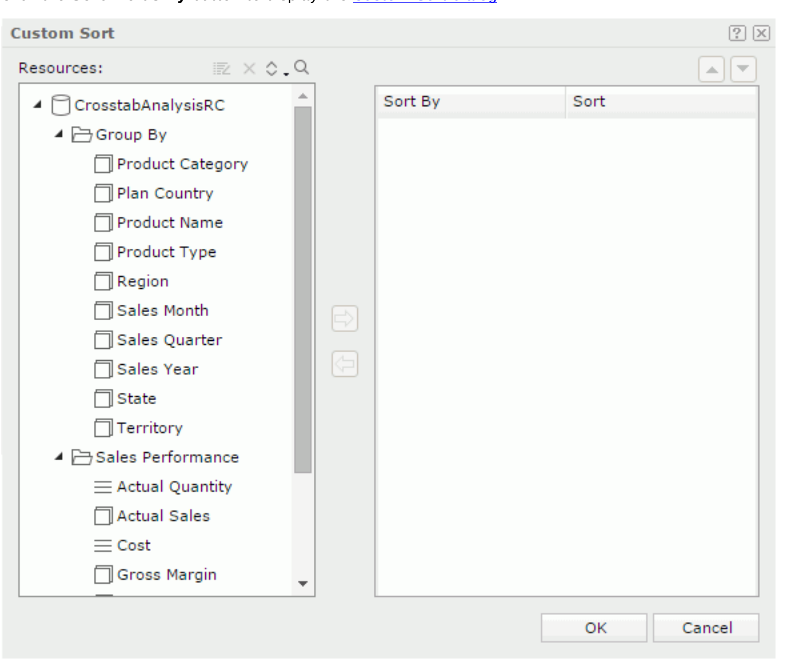
1. Locate the place in the report where you want to insert the table.
2. Drag **Table** from the Components panel to the destination. The Insert Table dialog appears.



3. Specify a title for the table in the Table Title text field, and if required, click [button] to set the font properties for the title.
4. From the Data Source drop-down list, select the business view in the current catalog, on which the table will be built. If required, click the **Filter** button to add some filter conditions to the business view to narrow down data displayed in the table.
5. Select the required table type: Group Above, Group Left, Group Left Above, or Summary Table.
6. In the Display tab, add the required fields from the Resources box to be displayed in the table. Specify the display name of any added field in the Label column if necessary. You can also specify how data in the table will be sorted.

   To set the sort manner:

   a. Click the **Sort Fields By** button to display the Custom Sort dialog.



   b. Select a field as the sort by field from the Resources box and click [button] to add it to the Sort By column.

   c. Select a sort order from the Sort column. Data in the table will then be sorted based on values of the specified sort by field in ascending or descending order.

   d. Repeat the above two steps to add more fields as the sort by fields if needed.

   e. Click [up] or [down] to adjust the order of the sort by fields.

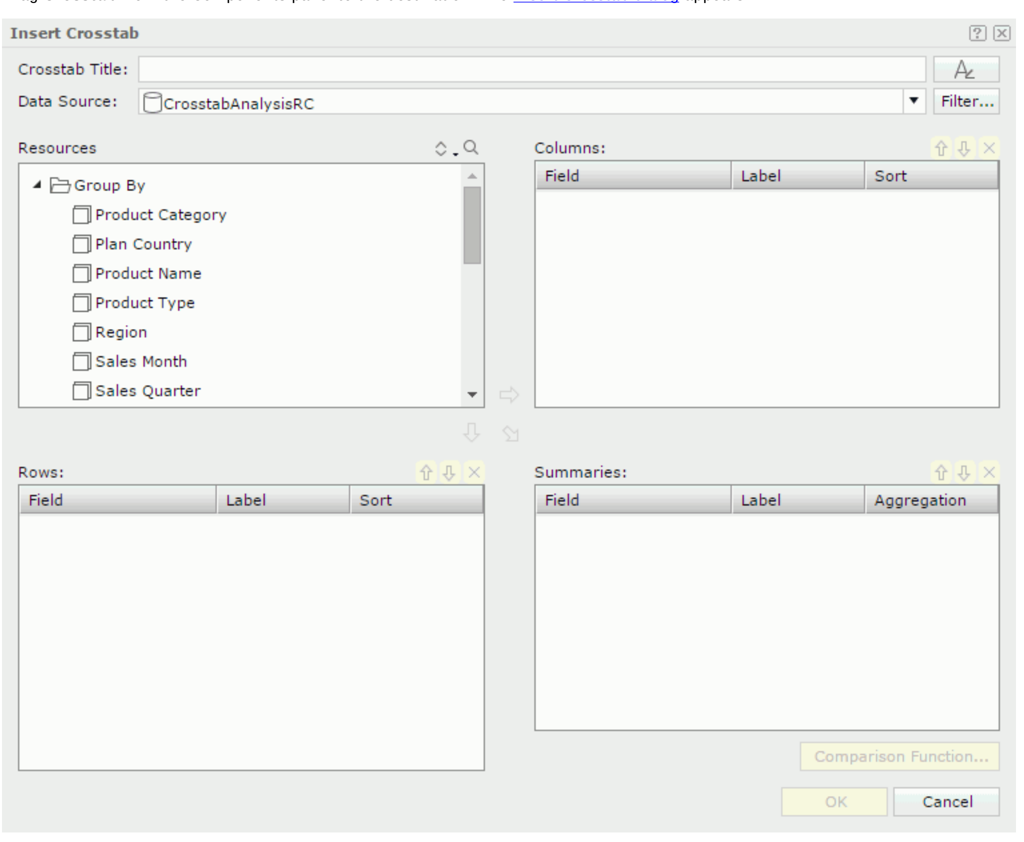
   f. Click **OK** to accept the sort criteria.

7. In the Group tab, add the group objects [icon] as the grouping criteria, then specify the sorting manner of each group in the Sort column, which can be No Sort, Ascend, Descend, or Custom Sort. When Custom Sort is selected, you can select some fields in the Custom Sort dialog, then the groups at the specfied group level will be sorted by the values of the first record in each group on the related fields.

   To adjust the order of the groups, select a group and click [up] or [down]
8. To add summaries, go to the Summary tab. Select the group to which the summary will be applied, then add an aggregation object [icon] as the summary field. For the Group Left table, you can use the Row and Column columns to control the position of the summary field in the table.
9. Click **OK** to insert the table.

Inserting a crosstab

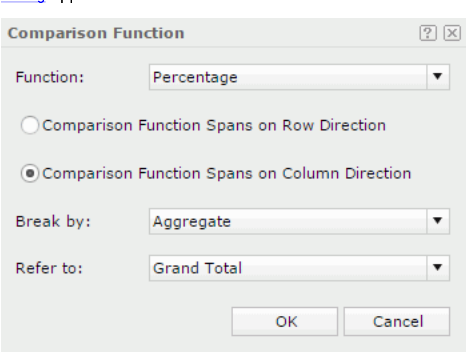
1. Locate the place in the report where you want to insert the crosstab.
2. Drag **Crosstab** from the Components panel to the destination. The Insert Crosstab dialog appears.



3. Specify a title for the crosstab in the Crosstab Title text field, and if required, click [button] to set the font properties for the title.
4. From the Data Source drop-down list, select the business view in the current catalog, on which the crosstab will be built. If required, click the **Filter** button to add some filter conditions to the business view to narrow down data displayed in the crosstab.
5. From the Resources box, select a group object [icon] and click [button] or [button] to add it to the Columns or Rows box as a group field. Then, in the Label column, edit the display name of the group object if required. This will label the row/column when the report is displayed. By default the Label column is blank and no label will be created for the row/column. In the Sort column, specify the sorting manner for the group field.
6. Select an aggregation object [icon] or a numeric detail object [icon] and click [button] to add it to the Summaries box as an aggregate field. If a detail object is added, specify the aggregate function for it in the Aggregation column. In the Label column, edit the display name of the aggregate field, which will label the summaries. If necessary, you can specify a comparison function for the aggregate field.

   To specify a comparison function for an aggregate field:

   a. Select the aggregate field in the Summaries box and click the **Comparison Function** button. The Comparison Function dialog appears.



   b. From the Function drop-down list, select the required function: Percentage, Permillage or Difference.

   c. Specify a position for the comparison function.
      - **Comparison Function Spans on Row Direction**
        The comparison function will be placed into the column total cell of the crosstab.
      - **Comparison Function Spans on Column Direction**
        The comparison function will be placed into the row total cell of the crosstab.

   d. Numbers that form the calculation of the comparison function are determined by the Break by and Refer to drop-down lists.

      Items in the Break by drop-down list vary with the position of the comparison function. It specifies the first parameter of the comparison function: the aggregate field or a subtotal.

      All available items are displayed in the Refer to drop-down list according to what you have selected from the Break by drop-down list. These items are outer group subtotals and the grand total. Select one as the other parameter of the comparison function.

   e. Click **OK** and you can see that a new field is added into the Summaries box. Set the display name for the field in the Label column as required.

7. Add more group/aggregate fields to the crosstab. If you want to remove any field, select it and click [×]. To adjust the order of the fields, select a field and click [up] or [down].
8. Click **OK** to insert the crosstab.

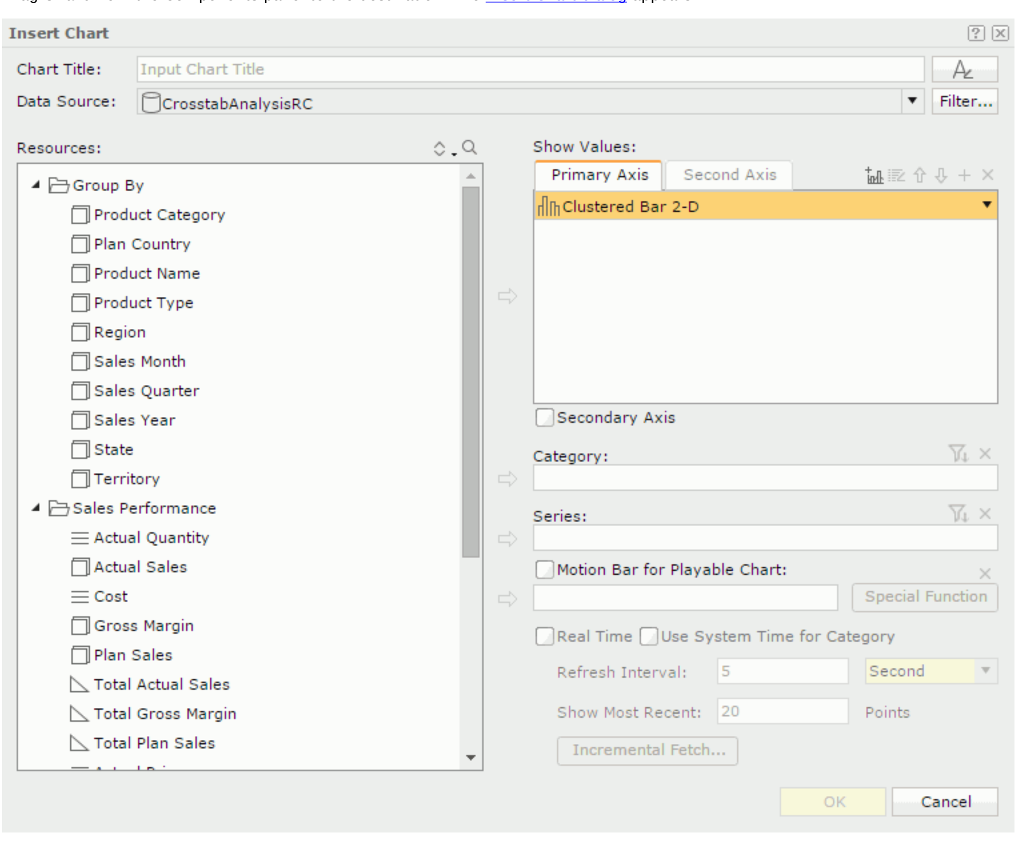
Inserting a chart

Normally, a chart displays values in a static way and you cannot change the values on it once it is created. However, JReport provides you with options to make the chart interactive and dynamic. For example, if your data source uses data that changes quickly over time such as stock market values, you can create a real time chart, so that the chart will update itself based on a defined interval by using the real time data from the data source. You can make a chart move at runtime based on the value changes of a motion field by creating a motion chart. In a motion chart, the chart is playable. You can start or stop the chart to play the dynamic trend of the motion field, control the moving speed of the chart, and if you create a bubble motion chart, you can even use a trail control to make the chart move showing a bubble or line trail.

In Web Report Studio, when you create a chart, you can choose to make it a common chart, an organization chart, a heat map, a real time chart, or a motion chart.

To create a common chart:

1. Locate the place in the report where you want to insert the chart.
2. Drag **Chart** from the Components panel to the destination. The Insert Chart dialog appears.



3. Specify a title for the chart in the Chart Title text field, and if required, click [button] to set the font properties for the title.
4. From the Data Source drop-down list, select the business view in the current catalog, on which the chart will be built. If required, click the **Filter** button to add some filter conditions to the business view to narrow down data displayed in the chart.
5. To create a single chart, in the Primary Axis box, select the required chart type from the chart type drop-down list.

   To create a combo chart, click [icon] above the Primary Axis box and an additional chart type will be added. You can replace the additional chart type by selecting the required one from the chart type drop-down list. Repeat this to add more chart types. Check the **Secondary Axis** checkbox if you want to have the secondary axis (Y2) and define the chart types on the axis as required. To delete a type, select it and click [×].
6. In the Primary Axis or Secondary Axis box, select a chart type and add an aggregation object [icon] or an additional value [#] as the data of the type. You can add more than one data field to a chart type. Each added chart type shall have at least one data field.

   If you select a bubble chart type, you need to specify the fields to be shown on the bubble X axis, Y axis and the value you want to show as the bubble radius in the Show Values box. Note that when you specify a value for the bubble X axis, this value will be displayed on the category axis instead of the one specified in the Category box. However, the value defined in the Category box will also be included in data calculation.

   To add an additional value to a chart type:

   a. Select the chart type in the Show Values box.

   b. In the Resources box, expand the **Additional Values** node, then select **Constant Value**/**Average Value**.

   c. Click [button] beside the Show Values box. The Edit Additional Value dialog appears.

   d. In the Name text box, specify the display name for the constant/average value.

   e. Input the constant value with numeric type in the Value text box, or select a field based on which the average value will be calculated from the Based On drop-down list.

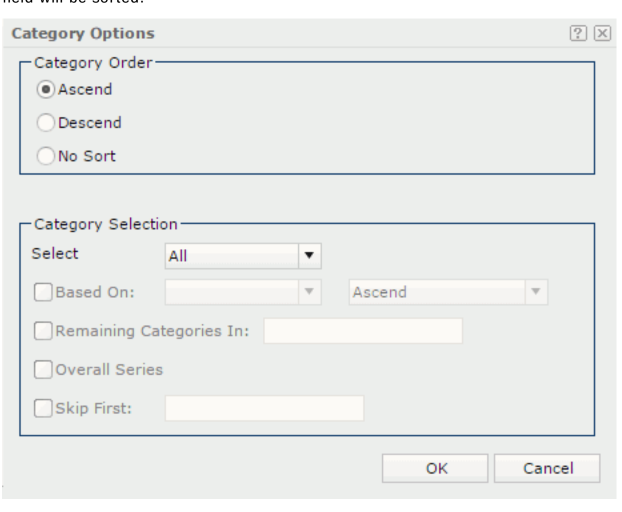
   f. Click **OK**, and the defined constant/average value will be added to the chart type.

      To modify a constant/average value, select the value in the Show Values box, then click [icon]. In the Edit Additional Value dialog, edit the value as required.

7. Select a group object [icon] in the Resources box and add it to the Category or Series box, the data of which will be displayed on the corresponding axis.
8. If you want to define some sort order and Select N condition on the category or series field, click [icon] above the Category or Series box, then define the order and condition in the Category/Series Options dialog.

   To define a sort order and Select N condition on the category/series field:

   a. In the Category/Series Order box of the Category/Series Options dialog, specify in which order values of the category/series field will be sorted.



   b. In the Category/Series Selection box, specify the Select condition to All, Top N or Bottom N. If All is selected, all category/series values will be shown in the chart; if Top N or Bottom N is selected, the text field next to it will be enabled and you can specify an integer here, which means that the first or last N category/series values will be shown in the chart.

   c. Check the **Based On** checkbox and specify values for the two drop-down lists that follow according to your requirement.

      If Based On is unchecked, the order of the first or last N category/series values will be based on what you specify in the Category/Series Order box of the dialog; if you check it, the order will be based on values of the summary field and the sort direction you specify in the drop-down lists next to Based On.

   d. If you have selected Top N or Bottom N from the Select drop-down list, you can check the **Remaining Categories/Series In** checkbox and then type a character string in the text field, so that the category/series values beyond the first or last N range will be merged into the group with the name as that character string.

   e. If you are specifying the Order/Select N condition for the category field, you can also check **Overall Series** to calculate the top or bottom N category values based on the series values.

   f. If necessary, you can check **Skip First**, and then input a number M in the text field to the right, then the first M category/series values will be skipped and the Select N condition will begin with M+1. The skipped values will be merged into the Remaining Categories/Series group.

   g. Click **OK** to accept the settings.

9. Click **OK** to insert the chart.

To create an organization chart:

Organization chart, also referred to as org chart, is a one-root-node-tree-structure diagram showing the ownership or reporting to relations among the nodes which are mapped to a specific entity.

1. Repeat the above steps 1 to 4 for creating a common chart.
2. In the Primary Axis box, select the Org chart type.
3. Select **Child** in the Org, next select a field from the Resources panel, and then click [button] to add the field for defining the entity.
4. Select **Parent** in the Org, next select a field from the Resources panel, and then click [button] to add the field for defining the ownership or reporting to relations among the entity members. For example, if the child field is Employee ID, the parent field can be the one about IDs showing which employ ID reports to which employ ID. Note that the parent field should use different one from the child field.
5. The properties panel displays a node model in the org chart. You can add objects including data objects, labels, and images from the Resources panel into the node by using the [button] button and then adjust their positions and sizes in the node and the size of the node if required. Those added objects will be displayed in each node as the information about the entity members.
6. Click **OK** to insert the chart.

To create a heat map:

Heat map is composed of rectangles marked by colors and sizes. The rectangles are grouped by group fields. Each rectangle represents a value of a group field or a combination of values of multiple group fields.

1. Repeat the above steps 1 to 4 for creating a common chart.
2. In the Primary Axis box, select the Heat Map type.
3. Add the fields used to group the data into the Groups box one by one. Use [up] and [down] to adjust the order of the groups. To remove a group field, select it and click [×].
4. If you want to define some sort order and Select N condition on a group field, click [icon] above the Groups box. In the Group Options dialog, specify the order and condition in the same way you do to the category/series field.
5. Add proper summaries into the Summaries box.

   The summaries should match the groups. For example, if the groups level is A > B > C, the static summaries grouped by C can be inserted into the Summaries box, but the static summaries grouped by A, B or other fields cannot.

   If no group is specified in the Groups box, you can insert any static summary. And its group-by field will be inserted into the Groups box automatically.
6. From both Groups and Summaries boxes, specify the fields to do color by and size by, and select the fields as label-by fields to display in the innermost rectangle in the heat map.

To create a real time chart:

Real time chart is supported on single bar, bench, line, and area chart types.

1. Repeat the above steps 1 to 4 for creating a common chart.
2. In the Primary Axis box, select a chart type of bar, bench, line, or area, then add the detail objects [icon] or group objects [icon] of numeric type as the data of the type.
3. Check the **Real Time** checkbox.
4. By default, Use System Time for Category is checked and you can see the text Use System Refresh Time is displayed in the Category box, which means the time at which the chart refreshes itself will be used as the category value. You can uncheck the Use System Time for Category option and add another group object [icon] to be displayed on the category axis. If you want to define some sort order and Select N condition on the category field you specify, click [icon] above the Category box, then define the order and condition in the Category Options dialog.
5. Specify the time interval at which the chart will get data and refresh itself automatically in the Refresh Interval text field.
6. Specify the most recent N records to be kept for the real time data on the chart in the Show Most Recent text field.
7. Click the **Incremental Fetch** button to add the fields you want to use as the unique key of the real time chart in the Unique Key dialog.

   Once a unique key is defined, each time when the real time chart automatically updates itself, duplicated data records will be filtered out based on the unique key. For instance, if you add the fields Country and Product ID as the unique key of a real time chart, when a record with the product ID 1 in USA has already been loaded into the chart, no more records of this product ID in USA will be added to the real time chart because they have the same unique key value.
8. Click **OK** to insert the chart.

To create a motion chart:

Motion chart is supported on single chart of bar, bench and bubble types.

1. Repeat the above steps 1 to 4 for creating a common chart.
2. In the Primary Axis box, select a chart type of bar, bench or bubble, then add the required aggregation objects [icon] or additional values [#] as the data of the type.

   If you select a bubble chart type, you need to specify the fields to be shown on the bubble X axis, Y axis and the value you want to show as the bubble radius in the Show Values box. Note that when you specify a value for the bubble X axis, this value will be displayed on the category axis instead of the one specified in the Category box. However, the value defined in the Category box will also be included in data calculation.
3. Select a group object [icon] in the Resources box and add it to the Category or Series box, the data of which will be displayed on the corresponding axis.
4. If you want to define some sort order and Select N condition on the category or series field you specify, click [icon] above the Category or Series box, then define the order and condition in the Category/Series Options dialog.
5. Check **Motion Bar for Playable Chart**, add a group object [icon] of Integer, Date or Time type as the motion field. When the element is of the Date data type, you can define some special function for it by clicking the Special Function button.
6. Click **OK** to insert the chart.

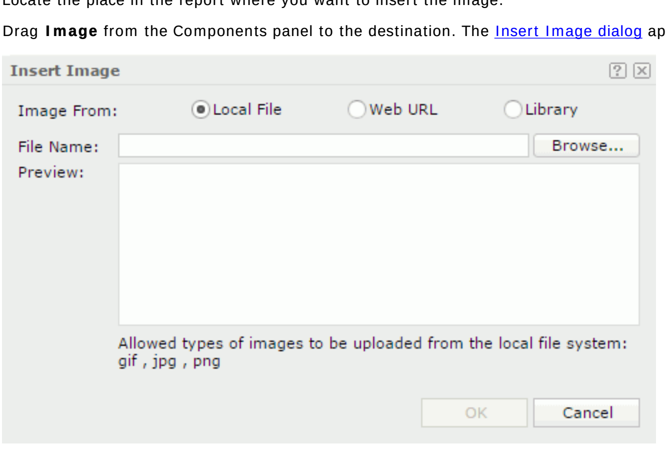
When a motion chart is created, you can use the motion control section to make the chart move. Click the play button and the chart will show its dynamic trend based on the value change of the motion field which is bound in the motion bar. To stop it, click the button again. You can also control its moving speed by dragging the slider between Slow and Fast on the speed control. For a bubble chart, you can control whether the chart will be moving in bubble or line trail.

Inserting a label

To insert a label into a web report, locate the place in the report where you want to insert the label, then drag **Label** from the Components panel to the destination. The label will then be inserted in the specified location.

Inserting an image

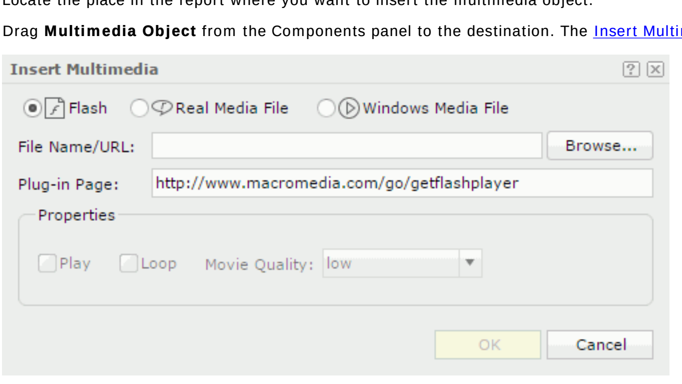
1. Locate the place in the report where you want to insert the image.
2. Drag **Image** from the Components panel to the destination. The Insert Image dialog appears.



3. Specify the image you want to insert.
   - To use an image in the local file system, select **Local File**, then click **Browse** to find the image.
   - To use an image on a website, select **Web URL**, then input the image URL or paste the URL in the Image URL text field.
   - To use an image in the image library of Web Report Studio, select **Library**, then select the image in the My Images box.
4. Click **OK** to insert the image.

Inserting a multimedia object

1. Locate the place in the report where you want to insert the multimedia object.
2. Drag **Multimedia Object** from the Components panel to the destination. The Insert Multimedia dialog appears.



3. Choose from the three multimedia object types: Flash, Real Media file, or Windows Media File.
4. In the File Name/URL text field, specify the full path of the multimedia object you want to insert or use the Browse button to find it if it is on your local disk. Or you can provide a URL for loading it from a website.
5. The Plug-in page text field provides a default URL from which to download the player to play the inserted multimedia object on a web page.
6. In the Properties box, specify the properties for the multimedia object as required.
7. Click **OK** to insert the multimedia object.

Inserting a web control

You can insert the following web controls into a web report: parameter control, parameter form control, filter control, and navigation control. For details, see Using web controls.

Inserting a special field

To insert a special field into a web report, just drag the special field from the Components panel to the destination in the report.

Note: When there are multiple data components in a web report, JReport will calculate the data and display the data components in an appropriate way to avoid missing data in the report as much as possible.

Making simple modifications to components

This section introduces the general actions that you can perform on the report components.

Resizing a component and its elements

To resize a component, click anywhere in the component, then you will see it is surrounded by a rectangle with resizing handles. Point to a handle, when the mouse pointer turns to a double-headed arrow, you can drag the handle to resize the component.

To adjust the width of a column in a table, click anywhere in the column, then click the button that appears on the column header to select the column. Point to the left or right boundary of the column, when the mouse pointer becomes a double-headed arrow, drag the handle to resize the column.

To adjust the row height in a table, click anywhere in the row, then click the button that appears at the leftmost on the row to select the row. Point to the upper or lower boundary of a row, when the mouse pointer becomes a double-headed arrow, drag the handle to resize the row height. Then all the other rows of the same role will be resized too. For example, if a detail row is resized, all rows in the detail area will be resized. If a group row is resized, all rows of the group will be resized, while the other groups' rows keep unchanged.

To resize the column or row in a crosstab, drag the right or lower boundary. Then all the columns or rows of the same role will change too.

For a tabular, point to the boundary between two cells and the mouse pointer will become a double-headed arrow, you can then drag the boundary to adjust the size of the related cells.

Hiding/Showing a component

To hide a component, right-click on the component, then click **Hide** on the shortcut menu.

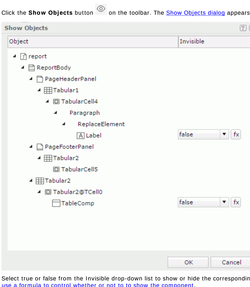
To hide a row or a column in a table, click anywhere in the row or column and you can see the buttons and appear at the leftmost on the row and on the column header. Click the corresponding button to select the row or column, then right-click and select **Hide** from the shortcut menu.

To show the hidden components, click **Menu** > **Edit** > **Unhide Components** and then select the desired components to show from the drop-down list.

You can also set the Invisible property in the Inspector panel to show or hide a component. To do this, select the node that represents the component in the report structure tree in the Inspector panel, then set its Invisible property as required. You can create an expression with the Formula Editor to control the property value by clicking in the value cell and selecting <Edit Expression> in the drop-down list. If the parent of the component has no data source, you can use a formula to control whether or not to show the component.

For any component whose parent doesn't have a data source, for example, a label in the tabular cell of a web report, you can also use the Show Objects dialog to show or hide them. To do this:

1. Click the **Show Objects** button on the toolbar. The Show Objects dialog appears.



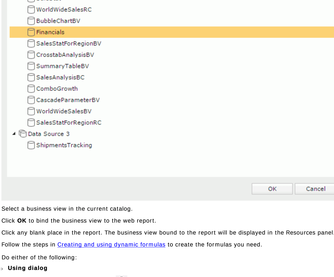
2. Select true or false from the Invisible drop-down list to show or hide the corresponding component. You can also use a formula to control whether or not to show the component.
3. When done, click **OK** to accept the settings.

Using formulas to control showing or hiding components

You can use formulas to control whether the components whose parents have no data source will be shown or not in a web report. However, before doing this, you need to first bind a data source to the web report, then create dynamic formulas of Boolean type based on this data source and use these formulas to control the Invisible property of the required components. A return value of true will hide the component.

To use formulas to control which components to show in a web report, follow the steps below:

1. Click **Menu** > **Edit** > **Bind Data**. The Bind Data dialog appears.



2. Select a business view in the current catalog.
3. Click **OK** to bind the business view to the web report.
4. Click any blank place in the report. The business view bound to the report will be displayed in the Resources panel.
5. Follow the steps in Creating and using dynamic formulas to create the formulas you need.
6. Do either of the following:
   - **Using dialog**
     a. Click the **Show Objects** button on the toolbar to display the Show Objects dialog.
     b. Click beside the Invisible property of the component which you want to control by formula and select the required formula from the drop-down list. Repeat this to select formulas for other components.
     c. Click **OK** to confirm the settings.
   - **Using the Inspector panel**
     a. Select the node that represents the component in the report structure tree in the Inspector panel.
     b. Click in the value cell of the Invisible property, then select the required formula from the drop-down list, or click <Create Formula> in the drop-down list and create a dynamic formula in the bound data source to control the property value.
     c. Repeat the above steps to specify the Invisible property of other components.

   Then whether the components will be shown or not will then be determined by the return value of the specified formulas.

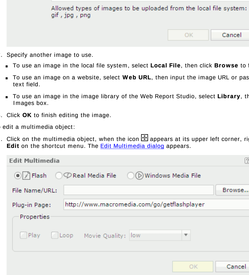
If you set the Invisible property of a component to true using a formula, the object will not be listed in the Menu > Edit > Unhide Components drop-down list. You can show it only by using the Show Objects dialog. Meanwhile, once the Invisible property of a component in a web report is controlled by a formula, the data source bound to the web report cannot be changed unless you remove the relationship between the formula and the property.

For a web report created in iReport Designer, if it has been bound with a data source before being published to iReport Server, and some dynamic formulas have been created based on this data source:

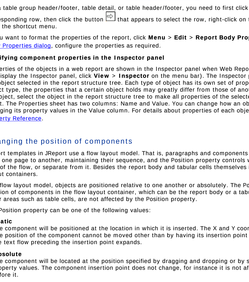
- If any of the formulas is used by the web report in iReport Designer, you cannot change the bound data source for the web report in Web Report Studio. However, you can use the formulas of Boolean type created in iReport Designer or create new formulas based on this data source to control the Invisible property in the Show Objects dialog.
- If none of the formulas is used by the web report in iReport Designer, you can change the bound data source for the web report in Web Report Studio as you want.

Editing a component

- To edit a label, click in the text and update the content. You can also use the Quick Formats toolbar to format the font, background color and alignment of a label.
- To edit a table, crosstab, or chart, use the corresponding report wizard on the shortcut menu. For details, see Manipulating data components.
- To edit an image:
  1. Click on the image, when the icon appears at its upper left corner, right-click on the icon and click **Edit** on the shortcut menu. The Edit Image dialog appears.



  2. Specify another image to use.
     - To use an image in the local file system, select **Local File**, then click **Browse** to find the image.
     - To use an image on a website, select **Web URL**, then input the image URL or paste the URL in the Image URL text field.
     - To use an image in the image library of the Web Report Studio, select **Library**, then select the image in the My Images box.
  3. Click **OK** to finish editing the image.
- To edit a multimedia object:
  1. Click on the multimedia object, when the icon appears at its upper left corner, right-click on the icon and click **Edit** on the shortcut menu. The Edit Multimedia dialog appears.



  2. Choose from the three multimedia object types: Flash, Real Media file, or Windows Media File.
  3. In the File Name/URL text field, specify the full path of the multimedia object you want to insert or use the Browse button to find it if it is on your local disk. Or you can provide a URL for loading it from a website.
  4. The Plug-in page text field provides a default URL from which to download the player to play the inserted multimedia object on a web page.
  5. In the Properties box, specify the properties for the multimedia object as required.
  6. Click **OK** to finish editing the multimedia object.
- For a tabular, you can edit it as follows:
  - **Merging tabular cells**
    Adjacent cells in a tabular which form a rectangle can be merged into one cell.
    To merge adjacent cells, select them one by one while holding the Ctrl key, then click **Menu** > **Format** > **Merge** or click on the toolbar, and these cells will be merged into one cell.
  - **Splitting a tabular cell**
    To split a tabular cell, select the cell and click **Menu** > **Format** > **Split**, then in the Split Cell dialog, specify the number of rows and columns and click **OK**. You can also click on the toolbar to split the cell vertically or horizontally.

Modifying component properties

You can modify the object properties with either the corresponding properties dialog or the Inspector panel.

Modifying component properties using the properties dialog

To edit the properties of an object in a report, right-click the object and select **Properties** from the shortcut menu. In the corresponding properties dialog, specify the settings as required. For detailed explanation about options in the properties dialogs, refer to the specific topics in Web Report Studio dialogs.

For a table group header/footer, table detail, or table header/footer, you need to first click anywhere in the corresponding row, then click the button that appears to select the row, right-click on the row and select **Properties** from the shortcut menu.

If you want to format the properties of the report, click **Menu** > **Edit** > **Report Body Properties**, then in the Report Body Properties dialog, configure the properties as required.

Modifying component properties in the Inspector panel

Properties of the objects in a web report are shown in the Inspector panel when Web Report Studio is in the edit mode (to display the Inspector panel, click **View** > **Inspector** on the menu bar). The Inspector panel lists all the properties of the object selected in the report structure tree. Each type of object has its own set of properties. Depending on the object type, the properties that a certain object holds may greatly differ from those of another. To edit the properties of an object, select the object in the report structure tree to make all properties of the selected object list in the Properties sheet. The Properties sheet has two columns: Name and Value. You can change how an object appears and behaves by changing its property values in the Value column. For details about properties of each object, see Web Report Object Property Reference.

Changing the position of components

Report templates in iReport use a flow layout model. That is, paragraphs and components in the report body can flow from one page to another, maintaining their sequence, and the Position property controls whether a component is to be part of the flow, or separate from it. Besides the report body and tabular cells themselves in iReport can also act as flow layout containers.

In a flow layout model, objects are positioned relative to one another or absolutely. The Position property controls the position of components in the flow layout container, which can be the report body or a tabular cell. Components placed in other areas such as table cells, are not affected by the Position property.

The Position property can be one of the following values:

- **Static**
  The component will be positioned at the location in which it is inserted. The X and Y coordinate properties are disabled. The position of the component cannot be moved other than by having its insertion point moved. This can happen when the text flow preceding the insertion point expands.
- **Absolute**
  The component will be located at the position specified by dragging and dropping or by setting its X and Y coordinate property values. The component insertion point does not change, for instance it is not affected by having text inserted before it.

To change the position of a component:

- Click on the component, when the icon appears at its upper left corner, right-click on the icon and on the shortcut menu, select the required item from the Position submenu.
- In the Inspector panel, select the node which represents the component, then set its Position property value as required.

Deleting a component

A component can be removed from the report if it is no longer required. To delete a component, right-click on the component and select **Delete** from the shortcut menu, or click on the component, when the icon appears at its upper left corner, right-click on the icon and on the shortcut menu click **Delete**. If a message prompts asking for your confirmation, click **Yes** to confirm the removal.

To delete a table column, click anywhere in the column, then click the button that appears on the column header to select the column. Right-click on the column and select **Delete** from the shortcut menu. In the Confirm dialog, click **Yes** to confirm the removal.

Note: In a web report, there must be one and only one tabular, so you cannot either insert another tabular or delete the current tabular.

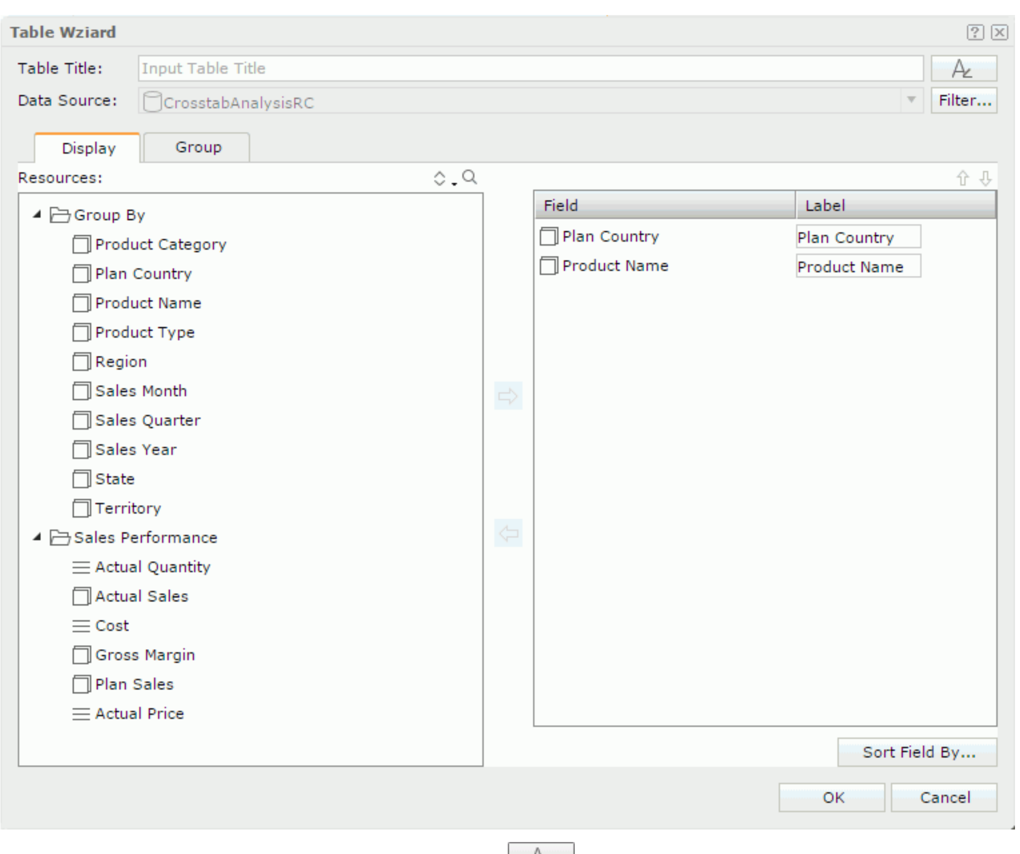
Manipulating data components

You can manipulate data components, which refer to tables, crosstabs, charts and geographic maps, in Web Report Studio as shown below. Note that, most of the manipulations require selecting the component first. To select a component, click anywhere in the component, when the icon appears at its upper left corner, click the icon.

Manipulating a table

- **Changing the table definition**
  1. Select the table and do one of the following:
     - Click **Menu** > **Edit** > **Wizard**.
     - Click the **Table Wizard** button on the Context toolbar.
     - Right-click the icon of the table and select **Table Wizard** from the shortcut menu.

     The Table Wizard appears.



  2. In the Table Title text field, edit the title of the table. You can click to customize the font, size, and style of the title.
  3. Click the **Filter** button to apply some filter conditions to narrow down data displayed in the table.
  4. In the Display tab, add or change the fields displayed in the table.
  5. In the Group tab, modify the grouping criteria of the table.
  6. Upon finishing, click **OK** to apply the modifications.

  For details about how to define a table, see Inserting a table.

- **Switching table column fields**
  In addition to the above method, another more convenient way to change the fields displayed in the detail columns or group columns of a table is by using the Switch command.
  - **To change the field in a detail column:**

    Click anywhere in the column, then click the button that appears on the column header to select the column. Right-click on the column and select **Switch Column** from the shortcut menu. The fields available for the detail column are listed in the submenu with the current used field checked. Select the field you want to use to replace the current one in the column.
  - **To change the field in a group column:**
    Right-click the group field and select **Switch Group** from the shortcut menu. The fields available for the group column are listed in the submenu with the current used field checked. Select the field you want to use to replace the current one in the column.
- **Adding or removing a table column or group by dragging**
  To drag a field into a table to become a table column, drag it from the Resources panel and then move the cursor on the column header to the border of an existing table column until a highlighted vertical line appears, then release the cursor, and the new column will be placed where the highlighted line lies.

  To drag a group field into a table to become a group, drag it from the Resources panel and then move the cursor in the detail section or to the border of an existing table group until a highlighted horizontal line appears, then release the cursor, and the new group will be placed where the highlighted line lies.

  To remove a table column, click anywhere in the column, then click the button that appears on the column header to select the column. Drag and drop the column to the Resources panel. Confirm the removal in the displayed dialog.

  To remove a group, click any value of the group, then click the button that appears at the leftmost on the row to select the group row. Drag and drop the row to the Resources panel. Confirm the removal in the displayed dialog.
- **Adjusting order of columns in a table**

  The order of columns in a table can be easily adjusted. To do this, click anywhere in a column, then click the button that appears on the column header to select the column. Drag the column to the left or right boundary of another column, when a highlighted line appears along the column boundary, release the mouse button, and you can see the order changes.
- **Adjusting the width of table columns according to contents**
  When the contents in cells of a table column need more space to completely display, you can adjust the width of the table column according to the contents. To do this, click anywhere in the column, then click the button that appears on the column header to select the column. Right-click on the column and click **Autofit** on the shortcut menu.
- **Aggregating on a detail column**
  You can summarize the data in a detail column. To do this:
  1. Do either of the following:
     - Right-click any value in the detail column and select **Aggregate On** from the shortcut menu.
     - Click anywhere in the column, then click the button that appears on the column header to select the column. On the Context toolbar, click the **Aggregate On** button .
  2. In the Aggregate On dialog, specify a function from the Function drop-down list to summarize the data.
  3. When done, click **OK**.
     - If the table has groups, you will find data in each group level and the whole table are summarized respectively in the column.
     - If the table has no groups, the summary will be based on the whole table.

  When you finish summarizing a detail column, you will find a dynamic aggregation is created at the same time which is given a default name Function_DetailFieldName in the Dynamic Resource > Aggregations list in the Resources panel and you can use it again in the current report if required.
- **Adding/Removing groups in a table**
  You can add more groups into a table or remove the groups that are not required from a table.
  - **To add a group into a table:**

    Select the table, then on the Context toolbar, click the **Add/Remove Group** button and you will get a drop-down list of fields in the business view that can be used as group by fields. From the list you can select the field you would like to add into the table as a group. If there is no existing group in the table, the added group will be placed at the left-above position. If the table already contains groups, the new group will be added as the highest level group and follow the same position pattern as the closest existing group.
  - **To remove a group from a table:**
    Right-click any value of the group and select **Delete** from the shortcut menu, then click **Yes** in the message dialog to confirm the removal. Or first select the table, then use the Add/Remove Group button on the Context toolbar: unselect the group you want to remove from the drop-down list, then click **Yes** in the message dialog.
- **Showing/Hiding detail columns**

  To show/hide a detail column, select the table, then on the Context toolbar, click the **Show/Hide Detail** button . From the drop-down list, select/unselect the field name to show/hide its detail column.

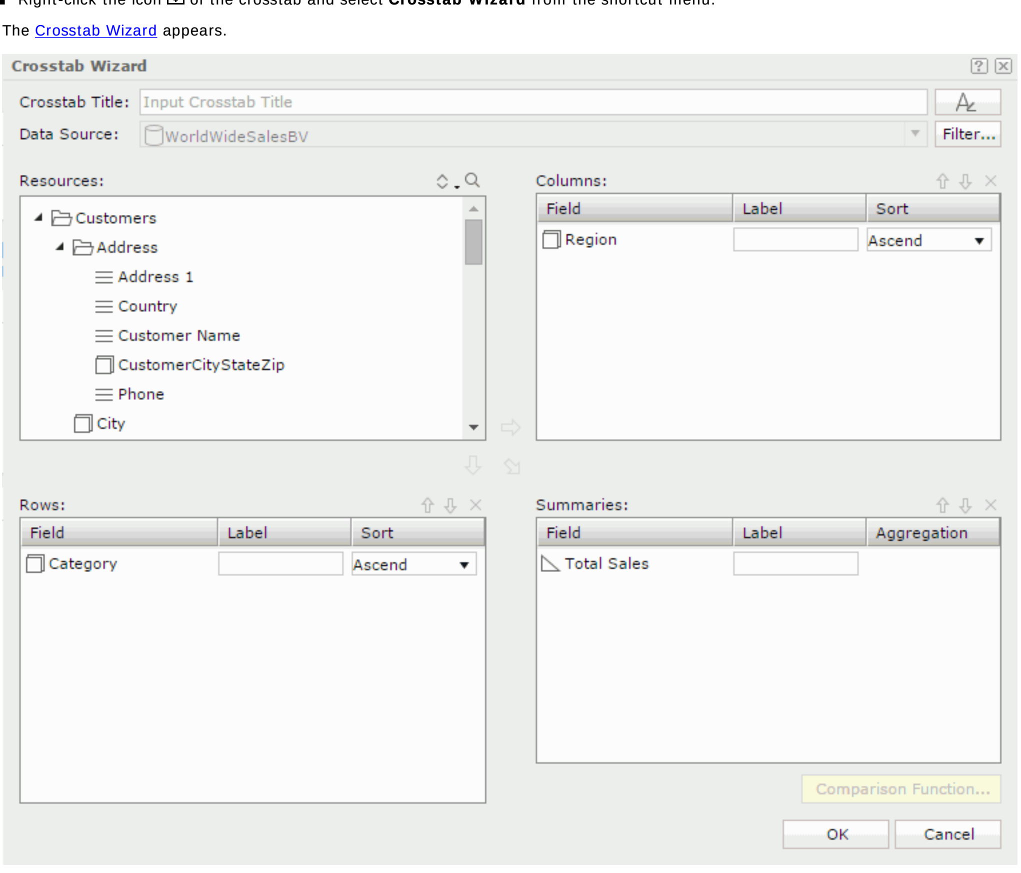
  You can also hide a detail column as follows: click anywhere in the column, click the button appearing on the column header to select the column, then right-click on the column and select **Hide**.
- **Showing/Hiding summaries**

  To show/hide a summary from a table, first select the table and then do either of the following:
  - On the Context toolbar, click the **Show/Hide Summary** button . From the drop-down list, select/unselect the summary field name to show/hide it.
  - Right-click the icon of the table, then on the shortcut menu, select/unselect the summary field name from the Show > Table Column submenu to show/hide it.
- **Customizing the field value data format**
  To customize the data format of the values of a field in a table, right-click on any value of the field, then select a format from the Format submenu, or input a format in the text box at the bottom of the submenu and click **Enter** on the keyboard.

Manipulating a crosstab

- **Changing the crosstab definition**
  1. Select the crosstab and then do one of the following:
     - Click **Menu** > **Edit** > **Wizard**.
     - Click the **Crosstab Wizard** button on the Context toolbar.
     - Right-click the icon of the crosstab and select **Crosstab Wizard** from the shortcut menu.

     The Crosstab Wizard appears.



  2. In the Crosstab Title text field, edit the title of the crosstab. You can click to customize the font, size, and style of the title.
  3. Click the **Filter** button to apply some filter conditions to narrow down data displayed in the crosstab.
  4. Change the fields and summaries used by the crosstab.
  5. Upon finishing, click **OK** to apply the modifications.

  For details about how to define a crosstab, see Inserting a crosstab.

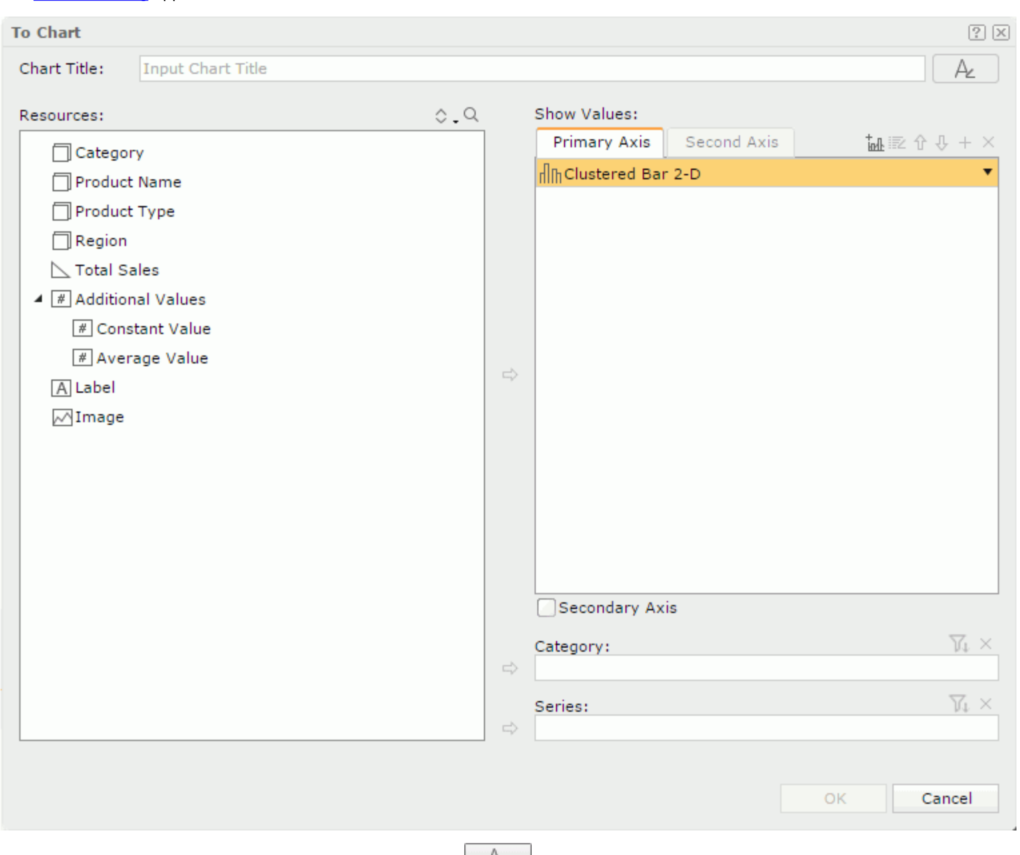
- **Switching crosstab fields**
  In addition to the above method, another more convenient way to change the fields in a crosstab is by using the Switch command. To do this, right-click a row/column/summary and select **Switch Row**/**Switch Column**/**Switch Summary** from the shortcut menu. The fields available for the row/column/summary are listed in the submenu with the current used field checked. Select the field you want to use to replace the current one.
- **Adding or removing row/column headers by dragging**
  To drag a group field into a crosstab to become a column header, drag it from the Resources panel and then move the cursor to a border of a column header until a highlighted horizontal line appears, then release the cursor, and the new column header will be placed where the highlighted line lies.

  To drag a group field into a crosstab to become a row header, drag it from the Resources panel and then move the cursor to a border of the row header until a highlighted vertical line appears, then release the cursor, and the new row header will be placed where the highlighted line lies.

  To remove a row/column header from a crosstab, first select it, next drag and drop it to the Resources panel, and then confirm the removal.
- **Inserting aggregate fields by dragging**
  You can drag and drop fields (detail fields, aggregation fields, and dynamic formulas/aggregations) from the Resources panel to a crosstab to add them as aggregate fields to summarize data in the crosstab. To do this, in the Resources panel, drag the required field and move the cursor above or under the desired subtotal or grand total cell until a highlighted horizontal line appears, then release the cursor.
  - If the selected field has been predefined with an aggregate function,
    - When you have specified any labels to label the columns/rows/summaries in the crosstab while creating it with the report wizard, the Insert Aggregation dialog appears. You can specify a label to label the newly created summaries if necessary. Click **OK** and the field is inserted to the specified position.
    - When the crosstab has no column/row/summary labels, the field is inserted into the specified position directly.
  - If the selected field has no predefined aggregate function, the Insert Aggregation dialog appears. From the Aggregate Function drop-down list, select the function to summarize the field, then input a label to label the newly created summaries if necessary. Click **OK** and the field is inserted to the specified position. However, whether the label you specify would be generated depends on if you have added any labels to label the columns/rows/summaries in the crosstab while creating it with the report wizard. If no label is defined in the report wizard, the label you add in the Insert Aggregation dialog will be ignored.

  When you have added an aggregate field using this way, you can find the new aggregate field in the Summaries box of the Crosstab Wizard.
- **Switching the aggregate function of an aggregate field**
  When an aggregate field in a crosstab has no predefined aggregate function, you can switch the function it uses at anytime. To do this, right-click the field and select **Switch Aggregate Function** from the shortcut menu, then choose a new function from the submenu.
- **Converting a crosstab into a chart**
  1. Select the crosstab and then do either of the following:
     - Click **Menu** > **Edit** > **To Chart**.
     - Right-click the icon of the crosstab and select **To Chart** from the shortcut menu.

     The To Chart dialog appears.



  2. In the Title text field, input a title for the chart. You can click to customize the font, size, and style of the title.
  3. The Resources box lists all the view elements used in the selected crosstab including group and aggregation objects. The chart can only be defined based on the view elements listed. Select the required chart type from the chart type drop-down list. Add a group object from the Resources box to the Category box, and so to the Series box, and aggregation objects to the Show Values box respectively.

     If you select a bubble chart type, you need to specify the fields to be shown on the bubble X axis, Y axis and the value you want to show as the bubble radius in the Show Values box. Note that when you specify a value for the bubble X axis, this value will be displayed on the category axis instead of the one specified in the Category box. However, the value defined in the Category box will also be included in data calculation.
  4. Click the **OK** button to finish the conversion.

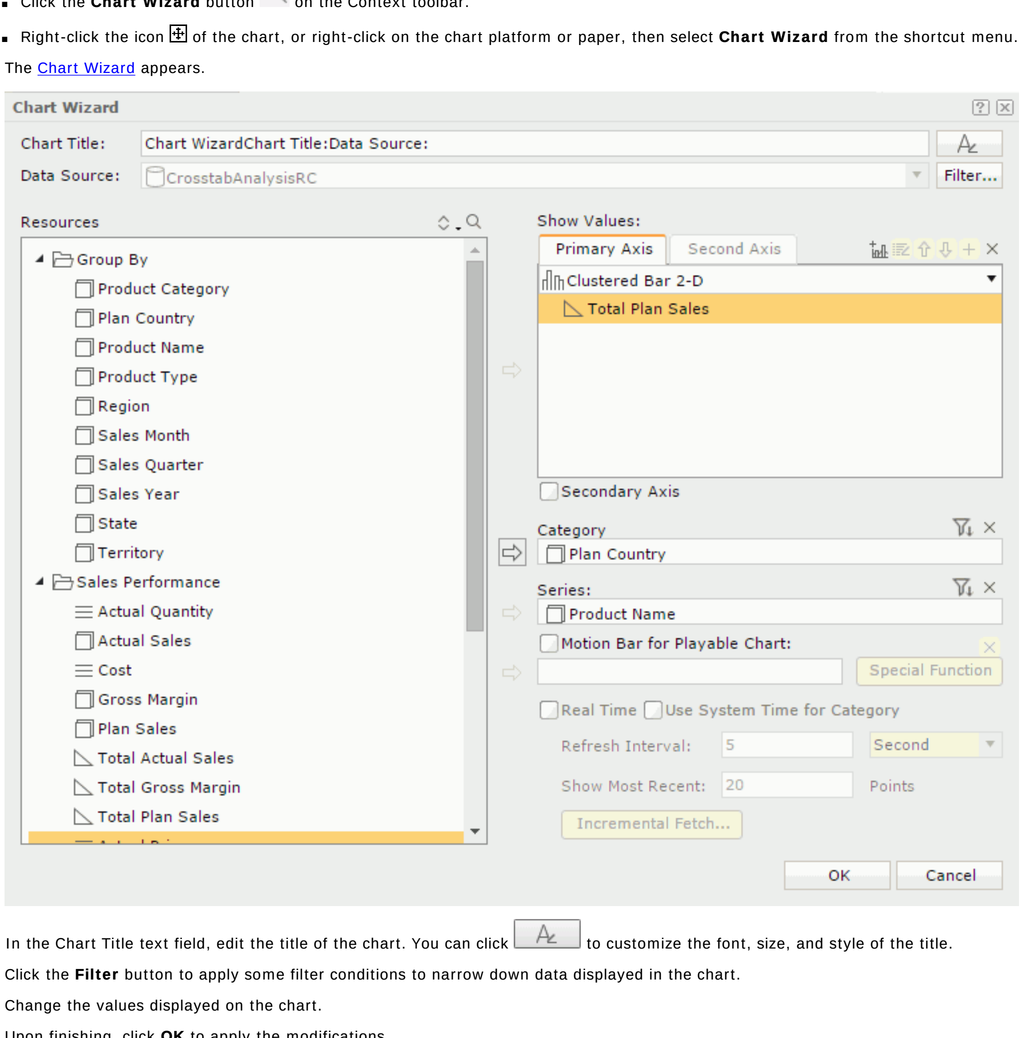
  For details about how to define a chart, see Inserting a chart.
- **Rotating a crosstab**
  Columns and rows in a crosstab can be exchanged. This operation is called rotating a crosstab.

  To rotate a crosstab, first select it, and then do one of the following:
  - Click **Menu** > **Edit** > **Rotate Crosstab**.
  - Click the **Rotate Crosstab** button on the Context toolbar.
  - Right-click the icon of the crosstab and select **Rotate Crosstab** from the shortcut menu.
- **Adjusting the width of crosstab fields according to the contents**
  When the contents in the field of a crosstab need more space to completely display, you can adjust the width of the field according to its contents. To achieve it, right-click the field and select **Autofit** from the shortcut menu.
- **Customizing the field value data format**
  To customize the data format of the values of a field in a crosstab, right-click on any value of the field, then select a format from the Format submenu, or input a format in the text box at the bottom of the submenu and click **Enter** on the keyboard.

Manipulating a chart

- **Changing the chart definition**
  1. Select the chart and then do one of the following:
     - Click **Menu** > **Edit** > **Wizard**.
     - Click the **Chart Wizard** button on the Context toolbar.
     - Right-click the icon of the chart, or right-click on the chart platform or paper, then select **Chart Wizard** from the shortcut menu.

     The Chart Wizard appears.



  2. In the Chart Title text field, edit the title of the chart. You can click to customize the font, size, and style of the title.
  3. Click the **Filter** button to apply some filter conditions to narrow down data displayed in the chart.
  4. Change the values displayed on the chart.
  5. Upon finishing, click **OK** to apply the modifications.

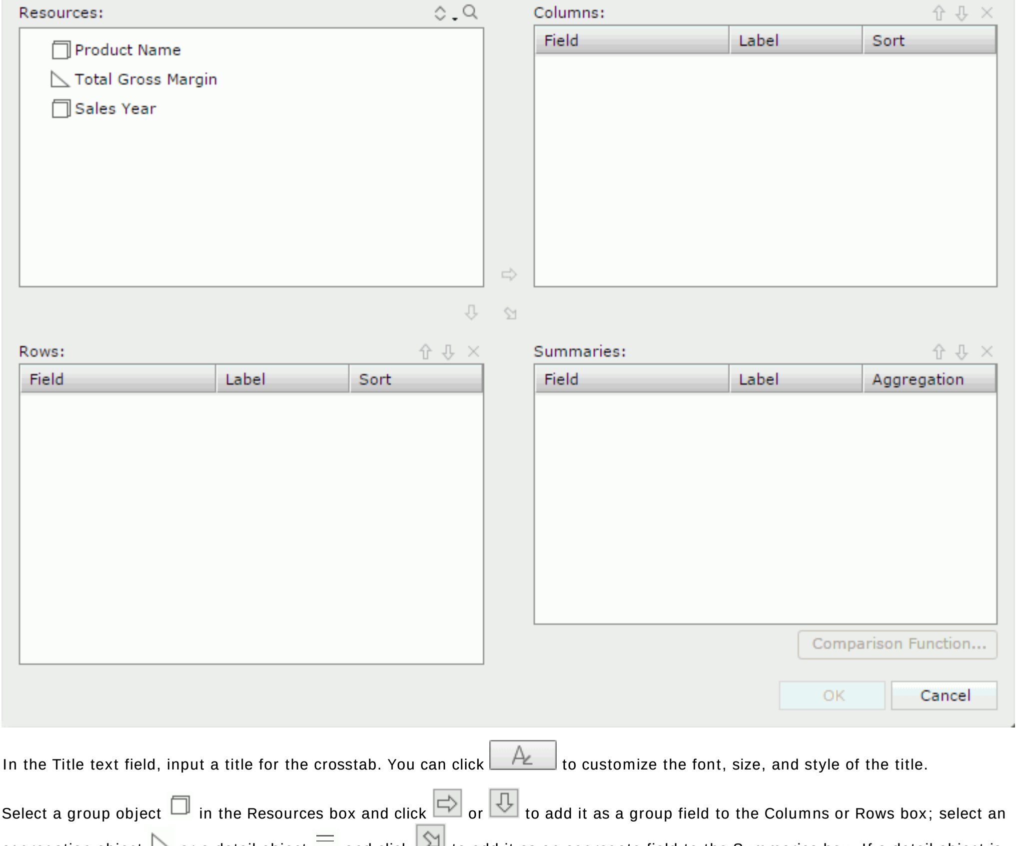
  For details about how to define a chart, see Inserting a chart.
- **Switching chart fields**
  In addition to the above method, another more convenient way to change the fields displayed on a chart is by using the Switch command.
  - **To change the field on the category/series axis:**
    Select the chart, right-click the icon of the chart or right-click on the chart platform or paper, then select **Switch Category**/**Switch Series** from the shortcut menu. The fields available for the category/series axis are listed in the submenu with the current used field checked. Select the field you want to use to replace the current one on the axis.
  - **To change the field on the value axis:**
    Select the chart, right-click the icon of the chart or right-click on the chart platform or paper, then select **Switch Value** from the shortcut menu. The fields available for the value axis are listed in the submenu with the current used field checked. Click an unchecked field to add it to the value axis.

    If a chart has multiple fields on its value axis, you can also click the up or down arrow beside the current used fields to adjust their order, or click a checked field to remove it from the chart. When there is only one field on the value axis, it cannot be removed.

    For a combo chart, you can change the fields displayed on the chart one by one for each chart type.
- **Formatting chart elements**
  You can format the chart graph, platform, paper, legend, X and Y axes, wall, floor, gridlines, and rectangles and rectangle titles of heat maps using the corresponding format command on the shortcut menu of a chart. For details about the element properties, refer to the specific format dialog in Web Report Studio dialogs.
- **Swapping chart groups**
  You can switch data between the category and series axes, or between the category and value axes of a chart if no field on the series axes.

  To swap the chart groups, first select the chart, then do either of the following:
  - Click the **Swap Chart Groups** button on the Context toolbar.
  - Right-click the icon of the chart, or right-click on the chart platform or paper and select **Swap Chart Groups** from the shortcut menu.
- **Converting a chart into a crosstab**
  1. Select the chart and then do either of the following:
     - Click **Menu** > **Edit** > **To Crosstab**.
     - Right-click the icon of the chart, or right-click on the chart platform or paper and click **To Crosstab** on the shortcut menu.

     The To Crosstab dialog appears.



  2. In the Title text field, input a title for the crosstab. You can click to customize the font, size, and style of the title.
  3. Select a group object in the Resources box and click or to add it as a group field to the Columns or Rows box; select an aggregation object or a detail object and click to add it as an aggregate field to the Summaries box. If a detail object is added, specify the aggregate function for it in the Aggregation column. Repeat this to add more group and aggregate fields.

     In the Label column, you can edit the label of a group field or aggregate field, and the Sort column allows you to specify a sorting manner on a group field.

     If necessary, click the **Comparison Function** button to set the comparison function for the aggregate fields.

     If you want to remove any group/aggregate field, select it and click .

     To adjust the order of group/aggregate fields, select a group/aggregate field and click or .
  4. Click **OK** to finish the conversion.
- **Showing/Hiding the chart legend or legend label**
  Select the chart, right-click the icon of the chart or right-click on the chart platform or paper, then select the corresponding Show or Hide command from the shortcut menu to show or hide the legend or legend label. However, the legend will always be displayed at the export or print result.
- **Changing chart type**

  Select the chart, then on the Context toolbar, click the **Chart Type** button . From the drop-down menu, select the desired chart type and its subtype.
- **Showing/Hiding labels on the X/Y axis**

  Select the chart, then on the Context toolbar, click the **Chart Options** button . From the drop-down menu, go to the **Label** submenu, then select/unselect the desired labels to show/hide them.
- **Showing/Hiding X/Y gridlines**

  Select the chart, then on the Context toolbar, click the **Chart Options** button . From the drop-down menu, go to the **Gridlines** submenu, then select/unselect the desired gridlines to show/hide them.

  When gridlines are shown, it is better to also have the wall shown so as to make the background gridlines more intuitive. To show the wall, follow the steps above, then on the Gridlines submenu, select **Wall**.
- **Changing the chart legend position**
  Chart legend can be placed at the top, bottom, left or right position in a chart. To change the legend position, select the chart, then on the Context toolbar, click the **Chart Options** button . From the drop-down menu, go to the **Legend** submenu and select the desired position.
- **Zooming in chart values**
  For a bar, bench, line, area or stock chart, you can select the values you are interested in to have them zoomed in. To do this, drag the mouse from the start value to the end value you want to select, then release the mouse. The selected values will be zoomed in. If you want to return to the initial status, just click the return button on the upper right corner of the chart paper.
- **Customizing the data format of chart data labels**
  You can customize the data format of the legend entry labels and data labels on the category and series axes if they are of Number or Date/Time type. To do this, right-click on any label and select the required format from the Format submenu, or input a format in the text box at the bottom of the submenu and click **Enter** on the keyboard.
- **Stopping or resuming a real time chart from refreshing**
  For a real time chart, you can stop or resume it from automatically refreshing by right-clicking it and selecting the Pause Refresh or Resume command from the shortcut menu.

Notes:

- The org chart type cannot be converted to any other chart type, and vice versa. Crosstabs cannot be converted to the org chart type, and vice versa.
- Additional values are supported only in chart. If you convert a chart with additional values into crosstab, the additional values are not converted together with the chart.
- You cannot zoom in chart values if your report is running in the Edit Mode of Web Report Studio. The action can only be performed in the View Mode.

Manipulating geographic map group markers/areas

- **Going up/down on geographic map group markers/areas**
  - For the group level that is higher than some other group levels in a geographic map component, point to its group marker/area, right-click it and select **Go Down** from the shortcut menu to jump one group level down.
  - For the group level that is lower than some other group levels in a geographic map component, point to its group marker/area, right-click it and select **Go Up** from the shortcut menu to jump one group level up.
- **Showing/Hiding geographic map group markers/areas**
  By default, all visible group markers/areas are shown. To hide them, right-click the geographic map (not the group markers/areas) and select **Hide Markers**/**Hide Areas** from the shortcut menu. If you want to show them again, right-click the geographic map and select **Show Markers**/**Show Areas** from the shortcut menu.

Binding links to objects

You can bind links to tables, images, DBfields, formula fields, parameter fields, special fields, and the category (C) sub labels, regardless of the data markers of charts. The link can either be a simple link or a conditional link. With conditional link, different targets can be loaded based on different conditions. However, conditional link is not supported on Mobile Dashboards and Mobile Report Viewer.

To bind a simple link to an object:

1. Right click the object and click **Link** on the shortcut menu to display the Insert Link dialog.
2. Select the Link type drop-down list, select the target to which the object will be linked. Report, URL, E-mail or Content.
3. Specify the options for the link target.
4. When done, click **OK** to save the link.

To bind a conditional link to an object:

1. Right-click the object and select **Link** from the shortcut menu to display the Insert Link dialog.
2. Check the **Conditional Link** check box.
3. Click the button ⁺ to open the Edit Conditions dialog to define a condition using either simple expressions or complex expressions according to your requirements.
4. Select a condition, which is displayed and highlighted in the conditions box in the Insert Link dialog.
5. From the Link Type drop-down list, select the target to which the object will be linked under this condition: Report, URL, E-mail or Content, then specify the options for the link target.
6. Repeat the above steps to add more conditions and define the link target for each condition.
7. To add a condition, select the condition in the Conditions box, then click ⁺. In the Edit Conditions dialog, with the expressions as required.
8. To remove a condition and the corresponding format, select the condition in the Conditions box and click ⁻.
   To adjust the priority of a condition, select the condition in the Conditions box and then click ↑ or ↓.
9. When done, click **OK** to finish defining the conditions.

For an illustrated example, see Example of binding a conditional link.

- To edit a link, right-click the object and select **Edit Link** from the shortcut menu, in the Edit Link dialog, with the link according to your requirement.
- To remove a link from an object, right-click the object and click **Remove Link** on the shortcut menu. In the warning message dialog, click **Yes** to confirm the removal.

Linking a report to another report

A report can be linked to another report, after which the trigger object in the primary report can be clicked in order to open the linked report. Select the primary report and then create the trigger object:

1. In the Insert Link dialog, select **Report** as the link type.

2. Click the **Browse** button beside the Report text field to specify the target web report you want as the linked report.
3. Specify the window or frame in which to load the linked report from the Target drop down list.
4. Click the **More** button to display more options regarding the link.
5. In the Filter tab, click ⁺ above the Components box to specify which components in the linked report will be coordinated with the current report.
6. Specify the conditions based on which the link relationship between the primary report and the selected components in the linked report is to be set:
   a. Select a component in the Components box.
   b. Click the button ⁺ above the Filter Conditions box and a condition line is added.
   c. Select a field from the drop down list in the Fields(Primary) column.
   d. Choose an operator from the drop down list in the OP column. The operator can be "=", "<", "<=", "!=", ">", ">=", or "IN".
   e. Select a field of the linked report from the drop down list in the Fields(Linked Report) column. All fields in the linked report are listed in the same order as the selected primary report field will be available.
   f. If necessary, you can specify more conditions by clicking the button ⁺ and then specifying the primary report field, the operator, and the corresponding field of the linked report. Note that the relationship between these conditions is AND, which means that (Report a)AND (Report b)AND (Report c)AND (Report d) all the conditions.
7. Set the Default Linked Component drop down list, select which component in the linked report will be interlinked with the primary report by default.
8. If the linked report uses parameters, go to the Parameters tab, the Target Report Parameters box lists the parameters of the linked report. You can assign fields of the primary report to the parameters. Then, when running the linked report from the link, the field values of the primary report will be assigned to the parameters automatically.
   Check on the Hide Linked Report Parameters check box, then the filter controls and editors in the primary report to be passed to filter objects in the linked report, go to the **Advanced** tab to set up the relationship between the filter objects as follows:
   a. Click ⁺ to add a line.
   b. In the Linked Report Parameters(Object) column, click ⁺ to select a filter object in the primary report in the Select Value dialog and click **OK**.
   c. In the Linked Report Properties(Object) column, click ⁺ to select a filter object in the linked report in the Select Value dialog and click **OK**. You need to make sure that values of the filter object in the primary report can be applied to the filter object in the linked report in the same line.
   d. Follow the above steps to set up the relationship between more filter objects.
9. Click **OK** to apply the settings.

Then, when you click the trigger object in the primary report, you will find that the linked report is displayed according to the specified conditions. The linked report will be opened when the primary report is opened in HTML, PDF or Excel format with the option Run Linked Report Automatically.

If the linked report is opened in the same frame as the primary report in Web Report Studio, you can click ⁺ on the toolbar to go back to the primary report, Click ⁻ next to ⁺ and you will get a drop-down list which lists the original primary report and the linked report. Click the primary report to go back to the primary report. Click the drop-down list to the current opened page, Click the linked report in the drop down list and you will be directed to that target.

Notes:

- When linking reports, you need to avoid link loops. For example, if you have linked report A to report B, then you should not link report B back to report A.
- The conditions specified in the Filter tab are used for filtering when the link is triggered or server runtime, that is, only the data that meet the conditions in the linked report will be displayed. However, when the link is triggered in HTML, PDF or Excel result, the conditions are used for setting up search criteria between the two linked objects, which means the linked report will be opened with the value of the primary report. The pages containing the data that meet the conditions in the linked report will be displayed.
- For web reports created in Report Designer which contain links, if the trigger objects of the links happen to be in the same page with the linked report, the linked report will be opened in the same frame as the primary report. In this case, the linked report will only display open the links in Web Report Studio only when the click prompt of the link action is specified to be the highest or report design time. If the priority of the drop down action is specified the highest, you can click the drop down icon, open the links only when you have reached the lowest level of the hierarchies by going down on the drop down.

Linking to a URL

1. In the Insert Link dialog, select **URL** as the link type.

2. Enter the URL in the Hyperlink text box directly. If needed, click the **Add Dynamic Field** button to insert a field into the URL to compose a dynamic URL.
   For example, you can compose a URL as follows: type in **http://www.google.com/search?q=** into the Hyperlink text box, then click **Add Dynamic Field** button, insert the field Country as the end of the URL. You can then click the label to open a specific URL that directs to a specific country.
3. Specify the frame from the Target drop down list.
   The Target property works when the link is triggered in Web Report Studio. When the link is triggered in Report Viewer, it opens in Report Frame, all the other targets are treated as New Window.
4. When done, click **OK** to set up the link.
   You can then click the trigger object to open the URL. The link is also available when the primary report is opened in HTML, PDF or Excel format.

Linking to an e-mail

1. In the Insert Link dialog, select **E-mail** as the link type.

2. Enter the e-mail address in the Hyperlink text box directly. If needed, click the **Add Dynamic Field** button to insert a field into the e-mail address to create a dynamic e-mail address.
3. When done, click **OK** to set up the link.
   You can then click the trigger object to open the e-mail with the information specified in the Hyperlink box. You can further customize the e-mail and send it. The link is also available when the primary report is opened in HTML, PDF or Excel format.

Linking to a Web data type field

Linking to a Web data type field is a Web data type field. The Web data type field will then be bound to the object and displayed as a hyperlink. You can open the Web content by clicking this hyperlink.

In Report, Web could be Image, Binary, Blob, Clob, LongBlob, LongClob, Varbinary, Longvarbinary, and so on.

To make an object linked to a Web data type field:

1. In the Insert Link dialog, select **Content** as the link type.

2. From the Query drop-down list, select the business view which contains the required Web data type field. Only query based business views that come from the same data source connection as the business view the current dashboard is based on are listed in the drop down list.
3. From the Content Type drop-down list, specify the content type for the Web type data. You can also click the button ⁺ to add a string type business view element or a record level (one row formula) in the selected business view to specify the content type dynamically.
4. In the File Name text field, specify a file name for the linked Web type data. You can also click the button ⁺ to add a string type business view element or a record level (one row formula) in the selected business view to control the file name.
   If you specify the file name and extension, choose a Blob data type field in the selected business view.
5. Click the **More** button to display more link options.
6. In the Filter tab, specify the filter conditions between the business view used by the current report and the business view that contains the linked Blob content as follows:
   a. Click ⁺ above the Field Conditions box and a condition line is added.
   b. Select the desired field of the business view used by the current report from the Fields(Primary) drop-down list.
   c. Choose an operator from the drop-down list in the OP column.
   d. Specify the field/formula of the business view that contains the linked Blob content from the drop-down list in the Fields(Linked) column. All fields that are of the same data type as the selected field in the Fields(Primary) column are available in the drop-down list.
   e. If necessary, you can specify more conditions by clicking the button ⁺ and then specifying the fields and operator. Note that the relationship between these conditions is AND, which means that (Report a)AND (Report b)AND (Report c)AND (Report d) AND (Linked Blob Type data which meets all of the conditions).
7. If the business view that contains the linked Blob content uses parameters, the Parameters tab will be enabled. In the Parameters tab, you can assign fields of the business view used by the current report to the parameters automatically. Check on the Hide Linked Report Parameters check box, the filter controls and editors in the linked report to be passed to filter objects in the linked report, go to the **Advanced** tab to set up the relationship between the filter objects as follows:
   a. Click ⁺ to add a line.
   b. In the Linked Report Parameters(Object) column, click ⁺ to select a filter object in the primary report in the Select Value dialog and click **OK**. You need to make sure that values of the filter object in the primary report can be applied to the filter object in the linked report in the same line.
   c. Follow the above steps to set up the relationship between more filter objects.
8. Click **OK** to apply the settings.

Then, when you click the trigger object in the primary report, you will find that the linked report is displayed according to the specified conditions. The linked report will be opened when the primary report is opened in HTML, PDF or Excel format with the option Run Linked Report Automatically.

Notes:

- When linking reports, you need to avoid link loops. For example, if you have linked report A to report B, then you should not link report B back to report A.
- The conditions specified in the Filter tab are used for filtering when the link is triggered or server runtime, that is, only the data that meet the conditions in the linked report will be displayed. However, when the link is triggered in HTML, PDF or Excel result, the conditions are used for setting up search criteria between the two linked objects, which means the linked report will be opened with the value of the primary report. The pages containing the data that meet the conditions in the linked report will be displayed.
- For web reports created in Report Designer which contain links, if the trigger objects of the links happen to be in the same page with the linked report, the linked report will be opened in the same frame as the primary report. In this case, the linked report will only display open the links in Web Report Studio only when the click prompt of the link action is specified to be the highest or report design time. If the priority of the drop down action is specified the highest, you can click the drop down icon, open the links only when you have reached the lowest level of the hierarchies by going down on the drop down.

Linking to a URL

1. In the Insert Link dialog, select **URL** as the link type.

2. Enter the URL in the Hyperlink text box directly. If needed, click the **Add Dynamic Field** button to insert a field into the URL to compose a dynamic URL.
   For example, you can compose a URL as follows: type in **http://www.google.com/search?q=** into the Hyperlink text box, then click **Add Dynamic Field** button, insert the field Country as the end of the URL. You can then click the label to open a specific URL that directs to a specific country.
3. Specify the frame from the Target drop down list.
   The Target property works when the link is triggered in Web Report Studio. When the link is triggered in Report Viewer, it opens in Report Frame, all the other targets are treated as New Window.
4. When done, click **OK** to set up the link.
   You can then click the trigger object to open the URL. The link is also available when the primary report is opened in HTML, PDF or Excel format.

Linking to an e-mail

1. In the Insert Link dialog, select **E-mail** as the link type.

2. Enter the e-mail address in the Hyperlink text box directly. If needed, click the **Add Dynamic Field** button to insert a field into the e-mail address to create a dynamic e-mail address.
3. When done, click **OK** to set up the link.
   You can then click the trigger object to open the e-mail with the information specified in the Hyperlink box. You can further customize the e-mail and send it. The link is also available when the primary report is opened in HTML, PDF or Excel format.

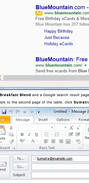
Linking to a Web data type field

Linking to a Web data type field is a Web data type field. The Web data type field will then be bound to the object and displayed as a hyperlink. You can open the Web content by clicking this hyperlink.

In Report, Web could be Image, Binary, Blob, Clob, LongBlob, LongClob, Varbinary, Longvarbinary, and so on.

To make an object linked to a Web data type field:

1. In the Insert Link dialog, select **Content** as the link type.



2. From the Query drop-down list, select the business view which contains the required Web data type field. Only query based business views that come from the same data source connection as the business view the current dashboard is based on are listed in the drop down list.
3. From the Content Type drop-down list, specify the content type for the Web type data. You can also click the button ⁺ to add a string type business view element or a record level (one row formula) in the selected business view to specify the content type dynamically.
4. In the File Name text field, specify a file name for the linked Blob type data. You can also click the button ⁺ to add a string type business view element or a record level (one row formula) in the selected business view to control the file name.
   If you specify the file name and extension, choose a Blob data type field in the selected business view.
5. Click the **More** button to display more link options.
6. In the Filter tab, specify the filter conditions between the business view used by the current report and the business view that contains the linked Blob content as follows:
   a. Click ⁺ above the Field Conditions box and a condition line is added.
   b. Select the desired field of the business view used by the current report from the Fields(Primary) drop-down list.
   c. Choose an operator from the drop-down list in the OP column.
   d. Specify the field/formula of the business view that contains the linked Blob content from the drop-down list in the Fields(Linked) column. All fields that are of the same data type as the selected field in the Fields(Primary) column are available in the drop-down list.
   e. If necessary, you can specify more conditions by clicking the button ⁺ and then specifying the fields and operator. Note that the relationship between these conditions is AND, which means that (Report a)AND (Report b)AND (Report c)AND (Report d) AND (Linked Blob Type data which meets all of the conditions).
7. If the business view that contains the linked Blob content uses parameters, the Parameters tab will be enabled. In the Parameters tab, you can assign fields of the business view used by the current report to the parameters automatically. Check on the Hide Linked Report Parameters check box, the filter controls and editors in the linked report to be passed to filter objects in the linked report, go to the **Advanced** tab to set up the relationship between the filter objects as follows:
   a. Click ⁺ to add a line.
   b. In the Linked Report Parameters(Object) column, click ⁺ to select a filter object in the primary report in the Select Value dialog and click **OK**. You need to make sure that values of the filter object in the primary report can be applied to the filter object in the linked report in the same line.
   c. Follow the above steps to set up the relationship between more filter objects.
8. Click **OK** to apply the settings.

Then, when you click the trigger object in the primary report, you will find that the linked report is displayed according to the specified conditions. The linked report will be opened when the primary report is opened in HTML, PDF or Excel format with the option Run Linked Report Automatically.

Notes:

- When linking reports, you need to avoid link loops. For example, if you have linked report A to report B, then you should not link report B back to report A.
- The conditions specified in the Filter tab are used for filtering when the link is triggered or server runtime, that is, only the data that meet the conditions in the linked report will be displayed. However, when the link is triggered in HTML, PDF or Excel result, the conditions are used for setting up search criteria between the two linked objects, which means the linked report will be opened with the value of the primary report. The pages containing the data that meet the conditions in the linked report will be displayed.
- For web reports created in Report Designer which contain links, if the trigger objects of the links happen to be in the same page with the linked report, the linked report will be opened in the same frame as the primary report. In this case, the linked report will only display open the links in Web Report Studio only when the click prompt of the link action is specified to be the highest or report design time. If the priority of the drop down action is specified the highest, you can click the drop down icon, open the links only when you have reached the lowest level of the hierarchies by going down on the drop down.

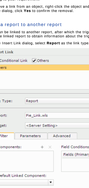
Example of binding a conditional link

This section provides an example to show the usage of conditional link and dynamic field.

1. Run the Link wizard in the Public Reports folder.
2. Click **Edit Route** on the toolbar.
3. We will define a link on the table of Report. Right-click on the Product Name value and then select **Link** from the shortcut menu.
4. In the Insert Link dialog, check the **Conditional Link** checkbox.
5. Click ⁺ to define a condition in the Edit Conditions dialog as follows:

6. For the product names other than Blue Mountain and Breakfast Blend, we will make them linked to an e-mail. Select the **Other** checkbox in the condition newest Others will be created.
7. From the Link Type drop-down list, select **E-mail**. In the Confirm dialog, click **OK**.
8. In the Hyperlink text box, type in **Dreamjobs.com** and then click the **Add Dynamic Field** button to insert the field Product Name right ahead of @.

9. We will make the field linked to a URL target based on the condition. From the Link Type drop down list, select **URL**. In the Confirm dialog, click **OK**.
10. In the Hyperlink text box, type in **http://www.google.com/search?q=** and then click the **Add Dynamic Field** button to insert the field Product Name at the end of the URL.



11. We will click the product names in the table to view the link result. Click **Blue Mountain** and a Google search result page with the key word Blue Mountain will be opened.

12. Click **Breakfast Blend** and a Google search result page with the key word Breakfast Blend will be displayed.
13. Navigate to the second page of the table, click **Sumatra** and an e-mail is displayed as follows:

Using dynamic resources

When you add fields to a report, sometimes you may find that the view elements that have been predefined in the business view cannot meet your requirements, in which case, you can create some dynamic resources and use them in the report to get the desired data. Then when you save the report, the dynamic resources will be saved along with the report as its resources.

Dynamic resources are report level resources, which means they are only available to the report for which they are created.

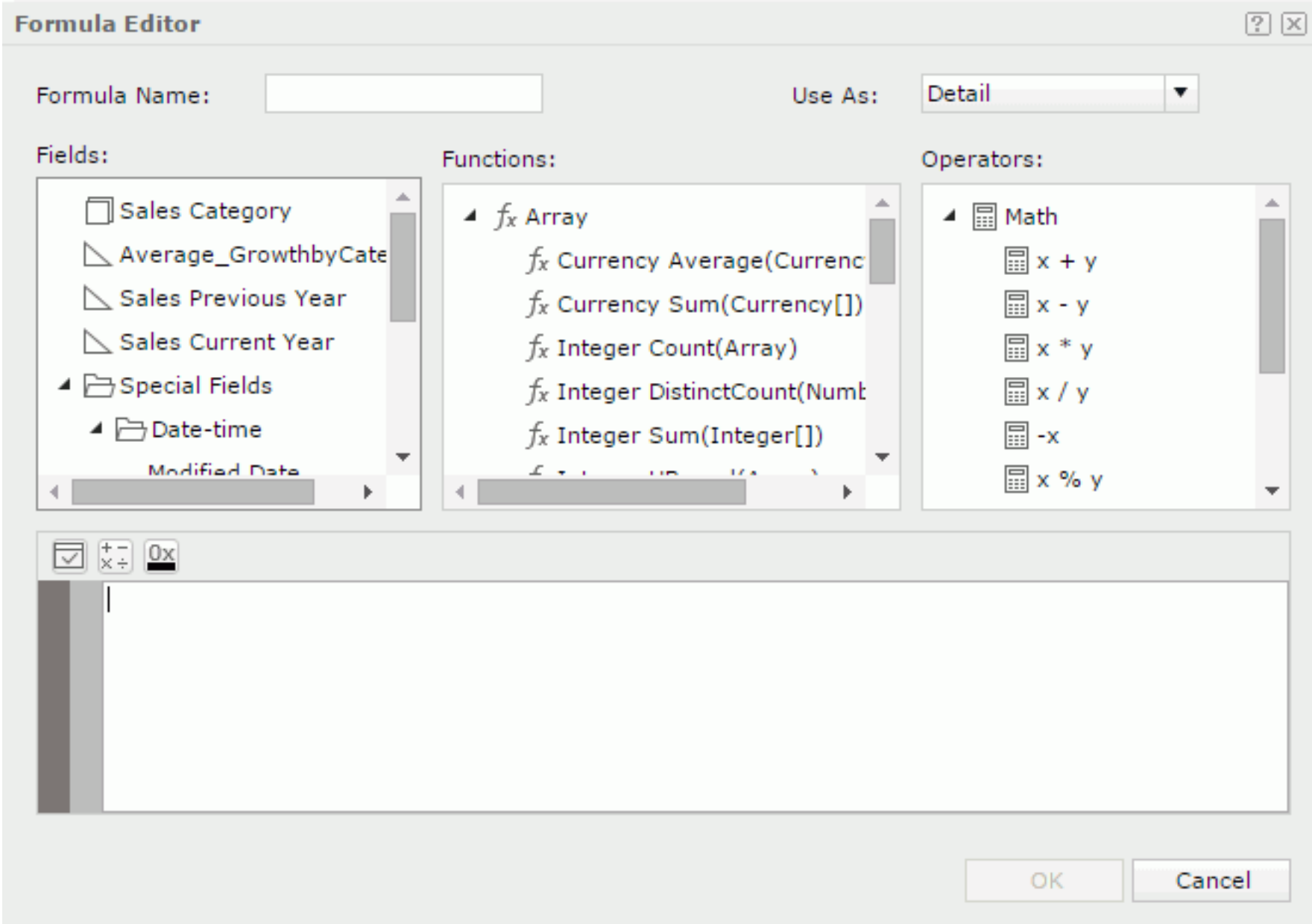
Dynamic resources that can be used in web reports include formulas and aggregations.

Creating and using dynamic formulas

You should have some knowledge of the formula syntax before you can successfully compose a formula with no errors. To learn the formula syntax, refer to Formula syntax.

To create a dynamic formula:

1. In the Resources panel of Web Report Studio or the Resources box of the Web Report Wizard, expand the **Dynamic Resource** > **Formulas** node, then click **<Add Formula•gt;**. The Formula Editor appears.



2. Enter a name for the formula in the Formula Name text field.
3. From the Use As drop-down list, specify what element type the formula is used as: Group, Detail or Aggregation.

   Which type a dynamic formula can be used as depends on the following:

   - Any formula can be used as Detail.
   - Record level pass 1 formula can be used as Group.
   - Comparable and record level pass 1 formula can be used as Group.
   - Group level formula can be used as Aggregation.
4. Compose the formula by selecting the required fields, functions and operators from the Fields, Functions and Operators boxes. You can also write the formula by yourself in the editing box.

   For details about the functions and operators, refer to Built-in functions and Operators.
5. Click the **Check** button to check whether or not the syntax of your formula is correct.
6. When done, click the **OK** button to create the formula.

Once a dynamic formula has been created, you can then drag it from the Resources panel to the desired position in the report as a detail/group or an aggregation object as defined, or use it when working with the report wizard.

In the Resources panel of Web Report Studio, you can make use of the shortcut menu of an existing dynamic formula to edit or delete the formula, or change the formula type as Group, Detail, or Aggregation if possible.

Notes:

- You can only use JDK (not JRE) to compile formulas created in JReport and save a dynamic formula with no errors into a web report.
- Currently, global variables are not supported in dynamic formulas.
- When formulas reference display names or mapping names, the names should not contain any of the following characters if the names are not quoted by double-quotation marks "":

  "~", "`", "!", "@", "#", "$", "%", "^", "&", "*", "(", ")", "-", "+", "=", "{", "}", "[", "]", "|", "\\", ":", ";", "\", " ' ", "<", ",", ">", ".", "?", "/"

  Examples:

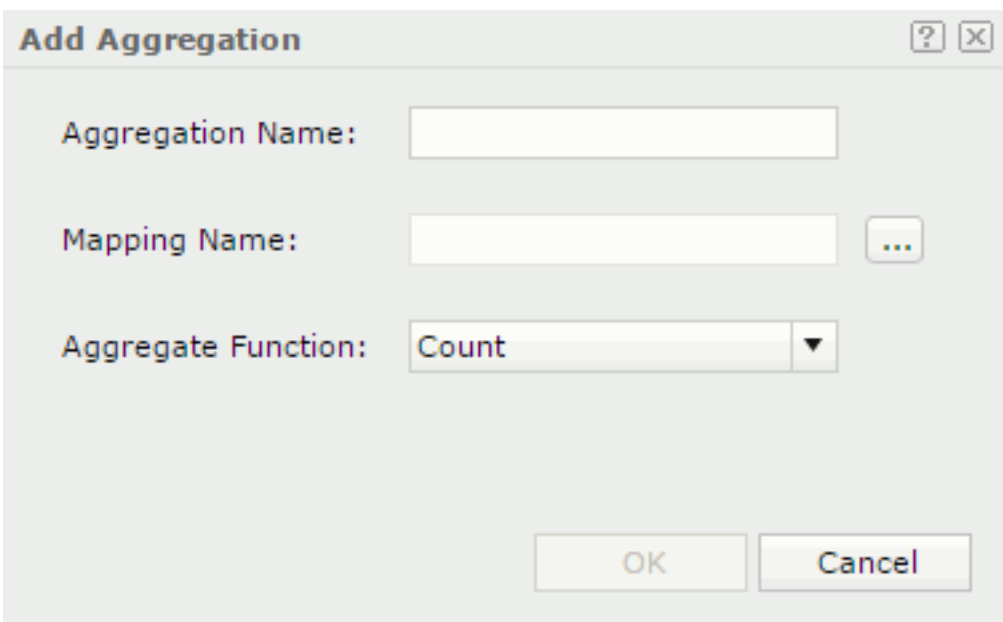
  - Expression @Customer#; will cause a syntax error. But @"Customer#" is ok.
  - If a field has the display name Category.Aggregation, when adding it to a formula, quote it as "Category.Aggregation" or "Category"."Aggregation".

Creating and using dynamic aggregations

In Web Report Studio, you can also create dynamic aggregations by mapping them to the available resources such as group objects, detail objects in the current business view and the dynamic formulas that have been created in the report.

To create a dynamic aggregation:

1. In the Resources panel of Web Report Studio or the Resources box of the Web Report Wizard, expand the **Dynamic Resource** > **Aggregations** node, then click **<Add Aggregation•gt;**. The Add Aggregation dialog appears.



2. In the Aggregation Name text field, specify the display name of the dynamic aggregation.
3. Click the button ... next to the Resource Name text field. In the Select Resource dialog, specify the field on which the dynamic aggregation is based.
4. From the Aggregate Function drop-down list, specify the aggregate function.
5. When done, click **OK** to create the dynamic aggregation.

Once a dynamic aggregation is created, you can then drag it from the Resources panel to the desired position in the report to see the desired data, or use it when working with the report wizard. And if you want to edit any dynamic aggregation or delete it, right-click the aggregation and click **Edit** or **Delete** on the shortcut menu.

Going through the report data

In a web report, you can choose to show certain groups of records according to your requirements, and switch among the groups to see the data you want:

- Go-up
  Enables you to obtain a different view of data by switching among groups.
- Go-to-by-value
  Enables you to filter data based on a go-to action to obtain a more detailed view of the data.
- Go-down
  Enables you to drill data to a lower-level group according to predefined hierarchies.
- Go-up
  Enables you to drill data to a higher-level group according to predefined hierarchies.
- Go-to-detail
  Enables you to concentrate on the details of a group.

After a "going" action has been performed, the data presented in the component will be re-loaded from the data buffer, showing only the data of the selected group, and the new report created can be viewed, printed, and exported to other formats in the same way as the original report.

Assume you have created a chart report of Clustered Bar 2-D type on the business view WorldWideSalesWV in Data Source 1 of the SamplePlanets catalog in the SamplePlanets folder, which shows product sales information with Region as the field on the category axis, Sales Year as the field on the series axis, and Total Sales as the field on the value axis, and applies the Neutral style to the chart. The chart shows as follows:

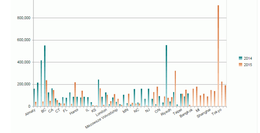
The business view WorldWideSalesWV has defined the following hierarchical relationship: Region > Country > State > City.

We will now take the chart as an instance to illustrate the going functions.

Go-to

The go-to action enables jumping to a different group by replacing the current. It is performed on chart categories and series.

1. Right-click any category of Region, of Asia-Pacific for example, and choose **Go To** from the shortcut menu. The list of groups available for "Go to" will appear on the submenu.



2. Click **State** on the submenu, then in the regenerated result, we can see that State becomes the group for categories and Sales Year remains the group for series.

3. To return to the original status, click the Undo button on the toolbar.

Go-to-by-value

The go-to-by-value action allows showing the information of another group while applying the current value being clicked on as a filter condition. It is performed on crosstab column/row headers, table groups, and chart categories and series.

1. Go back to the report in the above example.
2. Right-click the value **Asia-Pacific** of the Region group, and point to **Go to By Value** on the shortcut menu. A submenu for the command is displayed, which lists the same items as those of Go To.
3. Click **State** too and the result will be regenerated.

We can see that the result is different from that of go-to. This is because that, for the go-to-by-value action, the group of categories changes to State while being filtered by the Region value Asia-Pacific. That is, on the basis of the go-to action, a filtering action where Region = Asia-Pacific is performed at the same time, and thus the result of go-to-by-value is generated.

In addition, when a go-to-by-value action is performed, the "Go To" Filter panel will be displayed on the left of the Web Report Studio window, showing the group and the value that the filter is based on.

4. To go back to the original report, click the Undo button on the toolbar.

Go-down

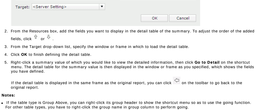
The go-down action allows going from a non-bottom level group to the one-level lower group while applying the current selected value as a filter condition. It is performed on table groups, crosstab column/row headers, and chart categories and series.

1. Go back to the original report in the above example, right-click the value **Asia-Pacific**, on the shortcut menu, point to **Go Down**, and we can see that Country is listed as the submenu item.

2. Click **Country** to see the result. It displays the data about countries in the Asia Pacific region.

3. The one-level lower group for Country defined in the hierarchy is State. Now click **China** directly and Jiangsuri will go down to State.

After these two go-down actions, we can see two filters are added in the "Go To" Filter panel, Region = Asia-Pacific and Country = China.



This is because, when you perform a go-down action, a filter will be created based on the value you click on. In this example, we first click on the Asia-Pacific region, so jiangsuri drills this region one level down to display countries in Asia-Pacific, and thus the filter Region = Asia-Pacific is created. If you want all data in the one level-lower group to be displayed when you drill down a group, you can remove the corresponding filter from the "Go To" Filter panel, by clicking **X** beside the filter condition.

Go-up

The go-up action allows jumping from a group which is non-top level in a predefined hierarchy to the one-level higher group. It is performed on table groups, crosstab column/row headers, and chart categories and series.

1. Based on the report result after go-down, right-click any value of State, Beijing for example, on the shortcut menu, point to **Go Up**, and we can see that Country is listed as the submenu item.

2. Click **Country** to see the result. Country is now the group for categories and the filter condition Country = China is removed from the "Go To" Filter panel.

3. The one-level higher group for Country defined in the hierarchy is Region. Now right-click any value of Country, Singapore for example, on the shortcut menu, point to **Go Up** and click **Region**, the chart restores to the original state.

Go-to-detail

The go-to-detail action is performed on the summary of the tables, crosstabs and charts. First define a table and make it contain the information you would like to view about the summary values. Suppose that the summary is total sales in different countries. Then when you perform go-to-detail action on the value of total sales in France, you will get the table displaying the fields you defined and the detail records of the condition Country=France. When you go to detail of the total sales in another country, the table will display the data of that country.

To define the detail table for a summary and perform the go-to-detail action on it:

1. Right-click any summary value (for a chart it is any data marker) and select **Edit Detail Table** from the shortcut menu. The Edit Detail Table dialog appears.

2. From the Resources box, add the fields you want to display in the detail table of the summary. To adjust the order of the added fields, click ↑ or ↓.
3. From the Target drop-down list, specify the window or frame in which to load the detail table.
4. Click **OK** to finish defining the detail table.
5. Right-click a summary value of which you would like to view the detailed information, then click **Go to Detail** on the shortcut menu. The detail table for the summary value is then displayed in the window or frame as you specified, which shows the fields you have defined.

   If the detail table is displayed in the same frame as the original report, you can click on the toolbar to go back to the original report.

Notes:

- If the table type is Group Above, you can right-click its group header to show the shortcut menu so as to use the going function. For other table types, you have to right-click the group name in group column to perform going.
- The Going feature is not supported on pg charts.
- You can also directly click group objects to perform the go-down action, however, if the group objects are also defined with some links.
- For web reports created in Report Designer, you can go down to the next level with a single click only when the click priority of the go-down action is specified to be the highest or second-highest design time. Otherwise, you have to use the Go Down command on the objects' shortcut menu to perform the action.
- For web reports created in Web Report Studio, you can only use the Go Down command on the objects' shortcut menu to perform the action. The link action takes the highest click priority by default and cannot be edited.

Applying filters

You can apply filters to business views and data components such as tables, crosstabs and charts of a web report so as to narrow down the data displayed in the web report.

Applying filters to business views

When creating web reports, you can choose to apply some filter to the specified business view to narrow down the data scope of the data component using the business view.

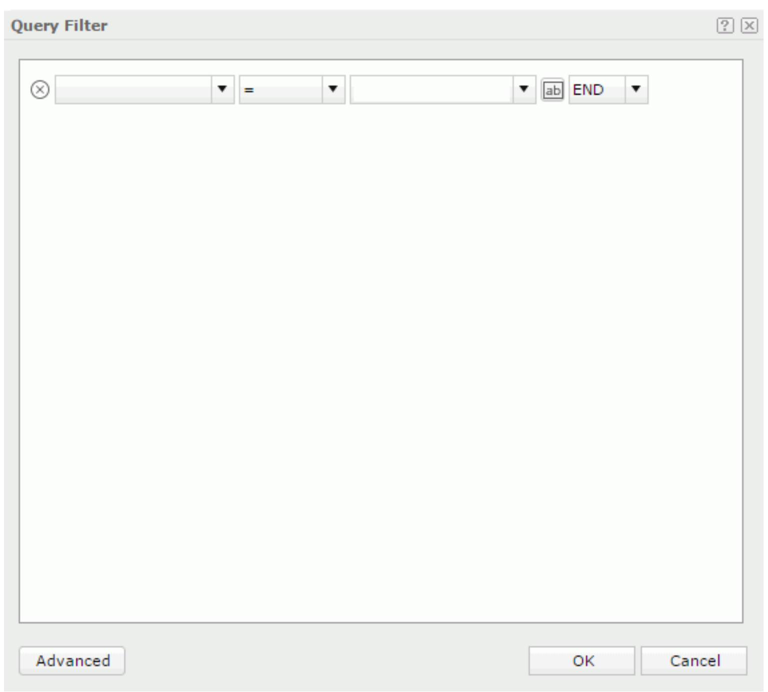
In Web Report Studio, filters for business views are defined into two categories: predefined filters and user defined filters. As the name suggests, predefined filters are defined in advance when creating or editing the business views in JReport Designer, and user defined filters are created on business views while they are used.

Filters can be applied to a business view in the report wizard.

1. In the report wizard, select the business view that you are going to add filters to from the Data Source drop-down list, and then click the **Filter** button on the right. The Query Filter dialog appears.
2. The dialog has the basic and advanced modes for you to define a filter using either simple expressions or complex expressions.

   When it is in the advanced mode, you can also choose to apply a predefined filter of the specified business view from the Query Filter drop-down list. If you prefer to define a filter on your own, select **User Defined** from the drop-down list, and then define the filter according to your requirements. You can also edit a predefined filter if required and save it as a user defined filter to the business view.

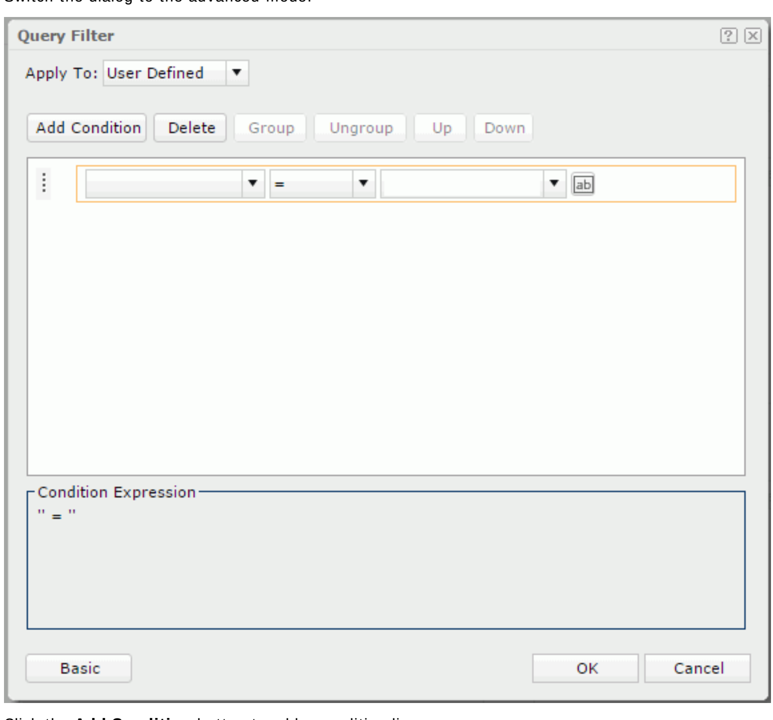
   - **To define a filter using simple expressions:**
     a. Make sure the dialog is in the basic mode.



     b. Select the field on which the filter will be based from the field drop-down list.
     c. From the operator drop-down list, set the operator with which to compose the filter expression.
     d. Type the values of how to filter the field in the value text box, or select one or more values from the drop-down list. You can also click the button and then select a parameter from the drop-down list to specify the value dynamically and when the available parameters cannot meet your requirement, you can create a local parameter to use in the filter.
     e. If you want to add another condition line, from the logic operator drop-down list,
        - To add a condition line of the AND relationship with the current line, select **AND**, then define the expression as required.
        - To add a condition line of the OR relationship with the current line, select **OR**, then define the expression as required.

     Repeat this to add more filter expressions if required. To delete a condition line, click on its left.

   - **To define a filter using complex expressions:**
     a. Switch the dialog to the advanced mode.



     b. Click the **Add Condition** button to add a condition line.
     c. From the field drop-down list, select the field on which the filter will be based.
     d. From the operator drop-down list, set the operator with which to compose the filter expression.
     e. Type the values of how to filter the field in the value text box, or select one or more values from the drop-down list. You can also click the button and then select a parameter from the drop-down list to specify the value dynamically and when the available parameters cannot meet your requirement, you can create a local parameter to use in the filter.
     f. To add another condition line, click the **Add Condition** button and define the expression as required. Then click the logic button until you get the required logic to specify the relationship between the two filter expressions. The logic can be AND, OR, AND NOT, or OR NOT.
     g. Repeat the above steps to add more filter expressions if necessary.

     To group some conditions, select them and click the **Group** button, then the selected conditions will be added in one group and work as one line of filter expression. Conditions and groups together can be further grouped. To take any condition or group in a group out, select it and click **Ungroup.** It is the equivalent of adding parenthesis in a logic expression.

     To adjust the priority of a condition line or a group, select it and click the **Up** or **Down** button.

     To delete a condition line or a group, select it and click the **Delete** button.

3. After you finish the report wizard, the specified filter is applied to the business view. If parameters are used in the filter conditions, they are listed in the Parameters panel and you can specify the parameter values in the panel to dynamically define the filter conditions.

Note: Query filters take effect on the component level, which means each time you create a component, you can apply a filter to the business view the component uses and it will not affect other components based on the same business view.

Filtering the data components in a report

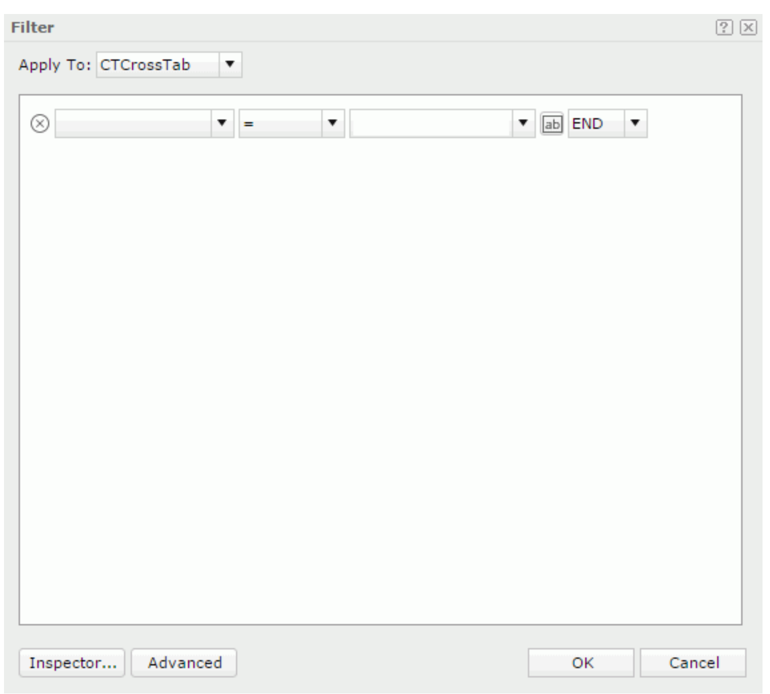
There are the following ways you can take in order to filter the data components in a web report.

Using the Filter dialog

When using the Filter dialog to filter report data, you can only make the filter applied to a specific data component in the current web report.

To filter report data using the Filter dialog:

1. Click **Menu** > **Edit** > **Filter**, or the **Filter** button on the Standard toolbar. The Filter dialog appears.



2. From the Apply to drop-down list, select the component in the web report to which you want to apply the filter.

   Tip: For web reports created in JReport Designer, the default selected component and available components in the Apply to drop-down list are determined by the Default for Filter and Invisible for Filter Dialogs properties of the components in the web reports.

3. Define the filter using either simple expressions or complex expressions.
4. When done, click **OK** to apply the filter. If parameters are used in the filter conditions, they are listed in the Parameters panel and you can specify the parameter values in the panel to dynamically define the filter conditions.

The Filter dialog provides an entry to all the filters used in the current web report. You can click the Inspector button to view the detailed filter information in the Filter Inspector dialog.

Using filter controls

You can also use the Filter web control to filter one or more data components that use the same data source in a web report. A filter control can do filtering based on one field. For details, see Using filter control to filter report data.

Using the Filter panel

The Filter panel on the left of Web Report Studio is used to filter data components in the current report that are using the same business view. To do this:

1. Add group and detail resources into the Filter panel by clicking **+** on the panel title bar. Each added group/detail and its values are housed in a separate box. Group and detail objects can be selected from the business views used by current report.
2. Select the values you would like to filter the report data. The selected values applies a filter condition to all the data components in the current report that are using the same business view, regardless whether the data components contain the fields holding those values.

   You can make use of the Ctrl or Shift key to do multiple selection.

The value selection applies a filter condition and the logic is as follows:

- **For one value selection:**

  Selected_Field=Selected_Value

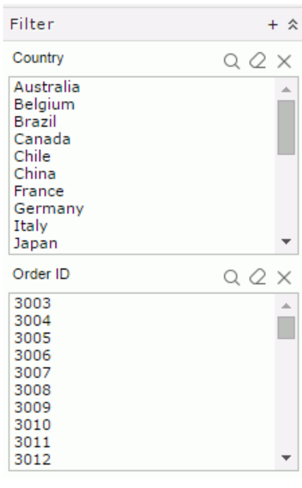
  For example, *Country=USA*

- **For multiple selection:**

  (Selected_Field1=Selected_Value1 or Selected_Field1=Selected_Value2) and (Selected_Field2=Selected_Value3 or Selected_Field2=Selected_Value4)...

  For example, *(Country=USA or Country=China) and (Year=2008 or Year=2009)*

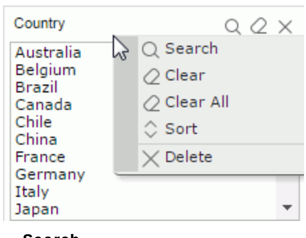
The following shows more about working with the Filter panel:



You can use the buttons on the bottom of the Filter panel to deal with the value selection in the panel.

- **Back**
  Goes back to the previous value selection status and refreshes the report data accordingly.
- **Clear**
  Removes all the value selection histories and all the filter conditions based on the selections, and refreshes the report data accordingly.
- **Forward**
  Goes forward to the next value selection status and refreshes the report data accordingly.

After right-clicking on a group/detail name title bar, these options are available for managing the group/detail object.



- **Search**
  Displays the quick search toolbar right above the value list which enables you to search values in the group/detail object. You can also use the button on the group/detail name title bar to launch the quick search toolbar. For detailed usage about the quick search toolbar, refer to Select Values dialog.
- **Clear**
  Cancels the selection of values in the group/detail object. You can also use the button on the group/detail name title bar to achieve this.
- **Clear All**
  Cancels the selection of all values in all the group and detail objects.
- **Sort**
  Sorts the values in the group/detail object in the ascending or descending order.
- **Delete**
  Removes the group/detail object from the Filter panel and the filter you created with the object will be removed from the report too. You can also click **X** on the group/detail name title bar to remove it.

Notes:

- When there are more than 300 values for an object, JReport will use Big Data Loading logic. In this case, the Shift Key for multiple selection will not work.
- The filters created via the Filter panel cannot be seen when web reports are opened in JReport Designer.

Cascading relationship between filters

The Filter panel can be regarded as a collection of special filter controls which apply to all data components using the same data source. While common filter controls can choose the data components they apply, still under the circumstance of using the same data source.

When there are filter controls, including the special ones in the Filter panel, that apply to the same data components, and when these controls' fields have cascading relationship, the cascading relationship will be revealed when you select values in the controls.

For example, there is a filter control based on the field Country, a filter control on City, and another on State. The first two share one table while the third shares nothing with the other two. In this case, Country and City values will show cascading relationship, but State values will not participate. You select USA in the Country filter control, the values in the City filter control will change as follows if the control has scrollbar: the cities belong to USA are displayed in the upper area of the filter control, and the other cities are put in the lower area and grayed out. For the case that the City filter control has no scrollbar: all the values remain their positions and the values not belonging to USA are grayed out. In both cases all the values are selectable. But the State values remain as before, since the selection of them will not affect the data components that the Country and City filter controls control.

Using the shortcut menu

You can use filter-related commands on the shortcut menu to filter the data in a table. To do this, point to any value of the field other than the group by field, by which you want to filter data, then right-click to show the shortcut menu. You will see the Filter item which provides a submenu containing the following commands:

- **Remove Filter**
  This command is enabled after you have applied filtering on the field. Clicking this item will remove all filters on this field.
- **First N**
  Shows the First N item with which you can filter data to display records that meet the First N condition. You can select a number from the submenu or enter a positive integer into the text box on the submenu to specify the First N condition.

  For example, if you select 5 from the First N submenu for a certain field, then only the records with the field value equal to one of the first five field values will be displayed.
- **Last N**
  Shows the Last N item with which you can filter data to display records that meet the Last N condition. You can select a number from the submenu or enter a positive integer into the text box on the submenu to specify the Last N condition.

  For example, if you select 5 from the Last N submenu for a certain field, then only the records with the field value equal to one of the last five field values will be displayed.
- **Field values**
  "Field values" is not the name for a command on the Filter submenu, but represents some items which are the values of the field you right-click on. Selecting any field value listed here will make the table only display records with the field value equal to the selected one.
- **More**
  This command is enabled if the Filter submenu cannot list all field values. When it is enabled, clicking it will show the Select Values dialog, with which you can search for and select values conveniently with the help of the quick search toolbar.

Using labels

For tables in web reports, you can use a label to control the filter conditions. This feature needs to be enabled at report design time.

1. In JReport Designer, select a label in a table in a web report, and then set its Filterable property to true.
2. Set the field by which you want to filter records as the value of the label's Bind Column property.
3. Save the report and publish it to JReport Server.
4. Run the report in Web Report Studio, and you can find that a filter button displayed beside the label. Click it to show the Filter list, which contains the same items as the Filter submenu. Click the corresponding item to filter the records.

   After a filter is applied on the field specified by the Bind Column property, the button will be highlighted. You can still click it to show the Filter list to change the filter condition or select the All item to remove the filter on the field.

Using server profile setting

You can also configure the server profile to enable filter records in tables by clicking the filter button on table column headers.

1. Open the Profile > Customize Profile > Common tab.
2. Check the option **Filter on Column Headers** to enable clicking the filter button on column headers to filter table records.
3. Check **Show Sort/Filter Status on Column Headers** if you want the the filter button to be always displayed on the table column headers with the current filter status. If this option is unchecked, each time you want to get the filter button for a column, you need to put the mouse pointer on its header to make the button shown.
4. Run a web report containing a table in Web Report Studio and you can click on the table column headers to filter data.

   To filter the table based on any column field, click on its column header to show the Filter list, which contains the same items as the Filter submenu. Click the corresponding item to filter the records.

   After a filter is applied on a field, the button on its column header will be highlighted. You can still click it to show the Filter list to change the filter condition or select the All item to remove the filter on the field.

Notes:

- When filtering the data in a table by either shortcut menu, label or column header, you may notice that the corresponding filter expression displays in the Filter dialog if you open this dialog.
- The Filter on Column Headers feature does not work on summary columns that contain multiple summary fields.
- When a table is designed with the Filter by Label feature in JReport Designer, if the Filter on Column Headers option is also enabled on the server profile, the settings defined in JReport Designer have the higher priority for the specified columns.
- For web reports created in JReport Designer, the default selected component and available components in the Apply to drop-down list of the Filter dialog are determined by the Default for Filter and Invisible for Filter Dialogs properties of the components in the web report.

Using web controls

Web controls are report components designed to be similar to the kinds of controls found on web pages. In Web Report Studio, these four types of web controls can be applied: parameter control, parameter form control, filter control, and navigation control.

This section describes each of the web controls and how to use them.

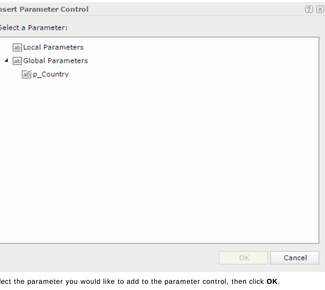
Using parameter control to specify a parameter to a report

A parameter control is a web control that is bound with a parameter used by the current report. By specifying values to the parameter in a parameter control, you can pass the parameter values to JReport and run the report with the specified values.

Cascading parameters cannot be used in parameter controls. If you want to do this, use parameter form controls instead.

To insert a parameter control and use it to specify a parameter to a report:

1. Drag **Parameter Control** from the Components panel to the destination in the report. The Insert Parameter Control dialog appears.



2. Select the parameter you would like to add to the parameter control, then click **OK**.
3. A parameter control will be added into the report. Specify the value for the parameter. You may specify the value in one of these ways.
4. Once the value in the parameter control changes, the report will rerun with the new parameter value.

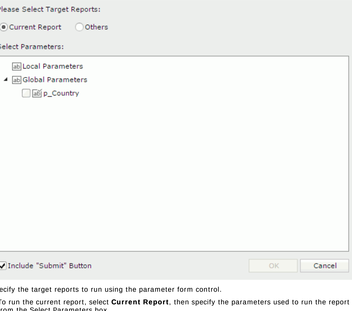
Note: If the specified parameter is no longer used in the report, the parameter control will become invalid.

Using parameter form control to run reports

A parameter form control is a web control that is bound with the parameters used by the current report or other reports. By specifying values to the parameters in a parameter form control, you can make the reports run with the specified parameter values.

To insert a parameter form control and use it to run reports:

1. Drag **Parameter Form Control** from the Components panel to the destination in the report. The Insert Parameter Form Control dialog appears.



2. Specify the target reports to run using the parameter form control.
   - To run the current report, select **Current Report**, then specify the parameters used to run the report from the Select Parameters box.
   - To run other reports, select **Others**, then select the reports you want to run. If all the selected reports contain no parameters, you cannot finish the dialog.
3. Specify whether to include the Submit button in the parameter form control. If Submit is included, it is used to submit the parameter values you specified in the parameter form control. If Submit is not included, once you change the values of a parameter in the parameter form control, the new values will be applied automatically.
4. Click **OK** in the dialog to save the changes.

   The parameter form control is now inserted in the report. It lists the selected parameters for the current report or lists all parameters used by the specified reports.
5. In the parameter form control, specify values of the listed parameters. You may specify the values in these ways.
6. Click the **Submit** button to run the current report or the specified reports if the button is available. If there is no Submit button, the change of values in the parameter form control will trigger report rerunning.

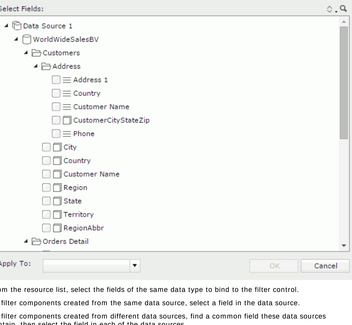
Note: If you save or publish a report containing a parameter form control to another directory, the reports that you bind the parameter form control with will not be saved or published along with the report.

Using filter control to filter report data

A filter control is used to filter one or more data components in a report, which refer to tables, charts, and crosstabs. For how a filter control works, see Filtering scenarios.

To insert a filter control and use it to filter report data:

1. Drag **Filter Control** from the Components panel to the destination in the report. The Insert Filter Control dialog appears.



2. From the resource list, select the fields of the same data type to bind to the filter control.

   To filter components created from the same data source, select a field in the data source.

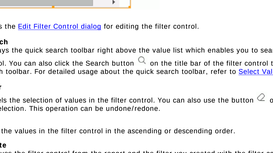
   To filter components created from different data sources, find a common field these data sources contain, then select the field in each of the data sources.
3. The Apply To drop-down list provides the components involving the selected fields. Select the components which you want to filter.
4. When done, click **OK**.

   The filter control is inserted in the report. It lists all values of the specified fields. You can select one or more values to apply.

After inserting filter controls in the report, you can also insert a navigation control for undoing/redoing the value selection in the filter controls. For details about the usage of navigation control, see Using navigation control to undo/redo value selection in filter controls.

Managing a filter control

After right-clicking on a filter control, these options are available for managing the filter control.



- **Edit**
  Opens the Edit Filter Control dialog for editing the filter control.
- **Search**
  Displays the quick search toolbar right above the value list which enables you to search values in the filter control. You can also click the Search button on the title bar of the filter control to launch the quick search toolbar. For detailed usage about the quick search toolbar, refer to Select Values dialog.
- **Clear**
  Cancels the selection of values in the filter control. You can also use the button on the title bar to cancel the selection. This operation can be undone/redone.
- **Sort**
  Sorts the values in the filter control in the ascending or descending order.
- **Delete**
  Removes the filter control from the report and the filter you created with the filter control will be removed from the report too. You can also use the button **X** on the title bar to delete the filter control.
- **Hide**
  Hides the filter control.
- **Properties**
  Opens the Filter Control Properties dialog for editing the properties of the filter control. The dialog provides an entry to changing the data fields that the filter control is based on and the application scope. To do this, in the General tab, click beside the Filter On text box. Then in the Select Field dialog, change the settings as you like and click **OK**.
- **Position**
  Specifies the position of the filter control.

Cascading relationship between filter controls

When there are filter controls that apply to the same data components, and when these controls' fields have cascading relationship, the cascading relationship will be revealed when you select values in the controls.

For example, there is a filter control based on the field Country, a filter control on City, and another on State. The first two share one table while the third shares nothing with the other two. In this case, Country and City values will show cascading relationship, but State values will not participate. You select USA in the Country filter control, the values in the City filter control will change as follows if the control has scrollbar: the cities belong to USA are displayed in the upper area of the filter control, and the other cities are put in the lower area and grayed out. For the case that the City filter control has no scrollbar: all the values remain their positions and the values not belonging to USA are grayed out. In both cases all the values are selectable. But that State values remain as before, since the selection of them will not affect the data components that the Country and City filter controls control.

Notes:

- When there are more than 300 values in a filter control, JReport will use Big Data Loading logic. In this case, the Shift Key for multiple selection will not work.
- Sometimes, when the position of a filter control is changed due to layout change, the advanced search toolbar in the filter control may not follow the filter control but stay where it is.

Using navigation control to undo/redo value selection in filter controls

A navigation control can be considered as an accessorial control for filter controls and used to deal with the value selection operations in all the filter controls in the same report.

To insert a navigation control into a report, drag **Navigation Control** from the Components panel to the destination in the report.

A navigation control is a combination of three buttons:

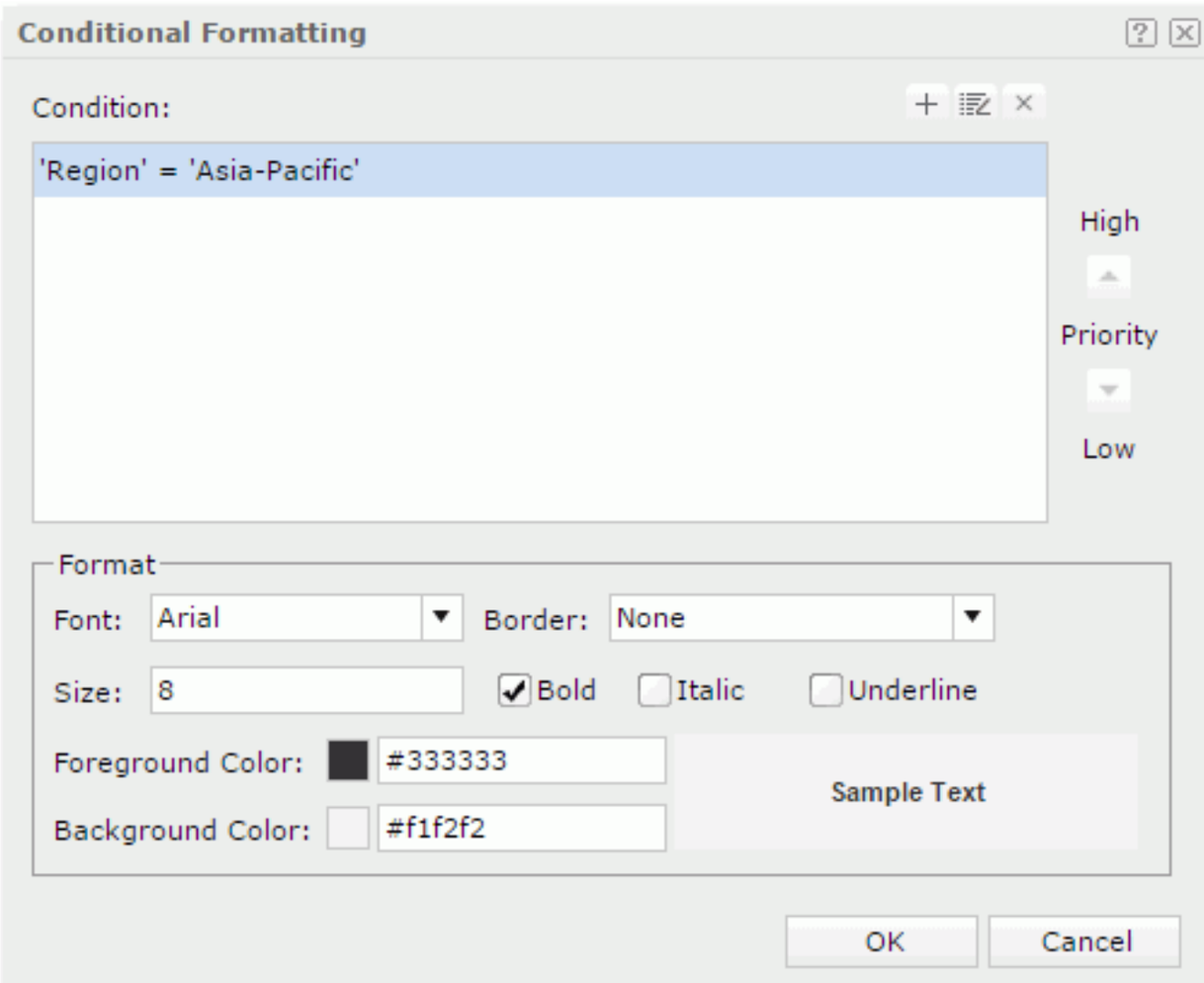
- **Back**
  Goes back to the previous value selection status and refreshes the report data accordingly.
- **Clear**
  Removes all the value selection histories and all the filter conditions based on the selections, and refreshes the report data accordingly.
- **Forward**
  Goes forward to the next value selection status and refreshes the report data accordingly.

Adding conditional formats to fields

You can add some conditional formats to a field in tables or crosstabs, which refer to the DBField, parameter field, formula field, and summary field, then when the specified condition is fulfilled, the defined format will be applied to the field values for highlighting.

To add conditional formats to a field:

1. Right-click the field and select **Conditional Formatting** from the shortcut menu. The Conditional Formatting dialog appears.



2. Click the button + to open the Edit Conditions dialog to define a condition using either simple expressions or complex expressions according to your requirements.

   The newly added condition will then be displayed and highlighted in the Condition box in the Conditional Formatting dialog.

3. In the Format box, set the format which will be applied to values of the field when the specified condition is fulfilled, for example, the font face, font size, font color, and so on.

4. Repeat the above steps to add more conditions and define the format for each condition.

   To edit a condition, select the condition in the Condition box, then click . In the Edit Conditions dialog, edit the expressions as required.

   To remove a condition and the corresponding format, select the condition in the Condition box and click ×.

   To adjust the priority of a condition, select the condition in the Condition box and then click ▲ or ▼.

5. Click **OK** to apply the conditional formats to the field.

Applying parameters

A parameter is a variable whose value is input at runtime. If a report contains a parameter, you need to specify a value for the parameter when running the report. The runtime parameter dynamically controls your report result.

The topic shows how to create and use local parameters in a web report and specify values for the parameters used in a web report.

Using local parameters

When working with a web report, sometimes you may find that the predefined parameters in the catalog cannot meet your requirements, in which case, you can create some local parameters and use them in the report to get the desired data. Then when you save the web report, the local parameters will be saved along with it as its resources.

Local parameters are report level resources, which means they are only available to the report for which they are created.

To create a local parameter:

1. Do one of the following:
   - In the report wizard, click the **Filter** button next to the Data Source text box, then in the Query Filter dialog, select the field to be filtered from the Field drop-down list, click the button in the Value text field and select **<Add Parameter...>** from the **Local Parameters** node of the drop-down list.
   - Click **Menu** > **Edit** > **Filter**, or the **Filter** button on the Standard toolbar to display the Filter dialog, select the component in the web report to which you want to apply the filter from the Apply to drop-down list, select the field to be filtered from the Field drop-down list, then click the button in the Value text field and selecting **<Add Parameter...>** from the **Local Parameters** node of the drop-down list.

   The Add Parameter dialog appears.
2. In the Name field, input a name for the parameter.
3. Select a parameter type from the Value Setting drop-down list. For parameters in XML connection, it can only be Type-in Parameter.
4. In the values section, specify the parameter values as required. The section varies with the type you select from the Value Setting drop-down list.
   - **For Type-in Parameter:**
     a. Specify the data type of the parameter value from the Value Type drop-down list.
     b. Click to add a value line, double-click in it and then type in a value of the specified data type.

        If the parameter is of Date, DateTime, or Time type, you can also click the calendar button to set a date and time value using either calendar or expression functions in the Calendar dialog. If you use an expression to specify the value, when you hover the mouse pointer over this value, a tip will appear showing its expression. Click on the value, the expression will be displayed in the value text box and you can edit the expression in the text box directly if you want. After editing, when you click elsewhere outside of the value text box:
        - If the edited expression is correct, a new value calculated by the expression will be displayed in the parameter value text box.
        - If the edited expression is wrong, no value will be created and the expression itself will be displayed and highlighted in red in the value text box. Correct the expression, or click the calendar button to specify a value.
     c. Repeat the second step to add more values.

        To adjust the order of the values, click or ; to remove any unwanted value, select it in the list and then click .
     d. Specify the value which will be set as the default selected value in the Enter Parameter Value dialog by selecting it from the list.
   - **For Bind with Single Column:**
     a. From the Source drop-down list, select the required business view in the current data source. Then, all fields in the specified business view will be available for value selection.
     b. Select a field from the Bind Column drop-down list. This field is used to filter the query when running a report with the parameter.
     c. If you want the values of another field to be displayed in the Enter Parameter Values dialog, which makes more sense to report end users than the values of the Bind Column field, select that field from the Display Column drop-down list.
   - **For Bind with Cascading Columns:**
     a. From the Data Source drop-down list, select the required business view in the current data source from which to get the columns.
     b. Click to add a parameter row.
     c. Click in the Bind Column cell and select a field from the drop-down list.
     d. If you want the values of another field to be displayed in the Enter Parameter Values dialog, which makes more sense to report end users than the values of the Bind Column field, click in the Display Column cell and then select that field from the drop-down list.
     e. Click in the Parameter cell to create the parameter. A name for the parameter will then be automatically added by JReport.
     f. Repeat steps b to e to create more parameters. Make sure that the selected Bind Column fields are of cascading relationship one level by one level down. In this way, a group of cascading parameters are created.

        To adjust the order of the parameters, click or ; to remove any unwanted parameter from the cascading group, select it and then click .
5. In the Options box, set options for the parameter according to your requirements.
6. When done, click **OK** to create the parameter.

   To further edit the parameter, right-click it and select **Edit** from the shortcut menu. Then in the Edit Parameter dialog, edit the parameter as required.

   To remove the parameter, right-click it and click **Delete** on the shortcut menu.

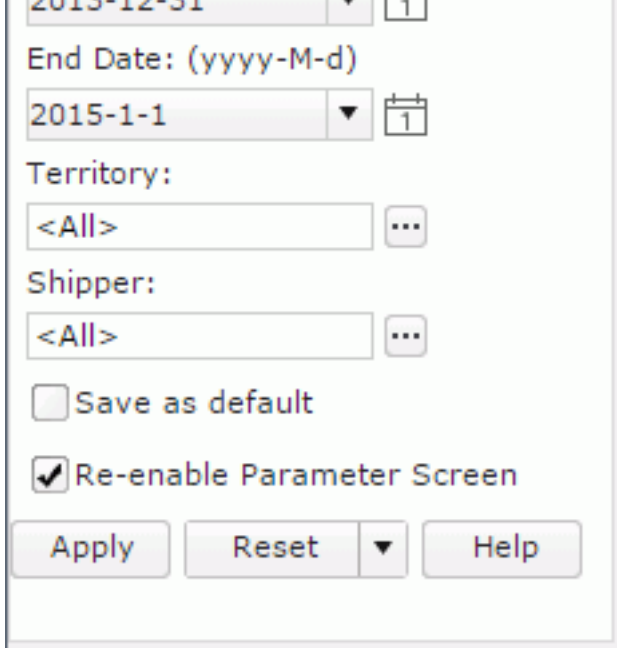
Once a local parameter is created, you can then use it to dynamically filter data of the web report, add it to a parameter control or parameter form control in the web report, or reference it in a dynamic formula used in the web report.

Specifying parameter values

When running a web report with parameters, a dialog is displayed for you to specify parameter values. After the report is opened in Web Report Studio, you can change the parameter values using the following ways.

Using the Parameters panel

The Parameters panel is available when the current web report uses parameters. It lists all the parameters used by the report with your last-time saved default values, which could be the values saved in this panel last time, or in the Enter Parameter Values dialog, Parameter Settings dialog, or when advanced running or scheduling the report. If you have not yet set the default values on the server, or if you did but your last-time saved default values cannot fully match the current parameters, all the parameters will use their default values specified in the parameters' definition as the initial values.



Edit the values according to your requirement and then click **Apply** to make the report run with the specified parameter values. You may specify parameter values in these ways.

If you want to save the current specified parameter values as the default values for the report, select **Save as default** in the panel (to make this option available, you need to make sure the Enable Setting Default Parameter Values For Web Report is selected in the Profile > Customize Server Preferences > Advanced tab).

If you would like the report to run on the console with the saved default values directly next time, without popping the Enter Parameter Values dialog, unselect **Re-enable Parameter Screen**, which is available when the option Enable Hiding Initial Parameter Dialog For Web Report is checked in the Profile > Customize Server Preferences > Advanced tab. Note that if later the last-time saved default values cannot completely match the report parameters, the parameter dialog will still be displayed. You can check this option to show the Enter Parameter Values dialog again after it has been hidden.

The above two options are user-report level settings, that is to say, they take effect when both the same user and report are matched. This also applies to admin users, and therefore admin cannot customize the settings for all users.

If you want to reset the parameter values, use the Reset button, which varies on different situations:

- When Save as default is available in the panel, the Reset button contains a text part and a triangle icon. You can choose to reset the values to either of the following by clicking the triangle. If you click the text part of the button directly, the values will be reset to the original default values.
  - **Original Default Values**
    The default values defined in the parameters' definition.
  - **User Defined Default Values**
    Your last-time saved default values.
- When Save as default is unavailable in the panel, the Reset button contains only the text part. Clicking it will reset the values to the original default values.

Using parameter controls

You can insert a parameter control and bind it with a parameter used by the current report. By specifying values to the parameter in the parameter control, you can pass the parameter values to JReport and run the report with the specified values. For details, see Using parameter control to specify a parameter to a report.

Using parameter form controls

You can insert a parameter form control, make it run the current report, bind it with one or more parameters used by the report. By specifying values to the parameters in the parameter form control, you can make the report run with the specified parameter values. For details, see Using parameter form control to run reports.

Sorting report data

You can make the data displayed in either descending or ascending alphabetical order in a report.

Sorting data in charts

Data labels on the category or series axes of a chart can be sorted alphabetically. To do this, select the chart, right-click the icon of the chart or right-click on the chart platform or paper, then on the shortcut menu, select **Ascend** or **Descend** from the Sort Category or Sort Series submenu.

To remove the sort condition, click **No Sort** on the Sort Category or Sort Series submenu.

Sorting data in crosstabs

You can make the records in a crosstab sorted based on any column or row field. To do this, right-click on any value on the column or row header, then choose the command **Sort** > **Ascend** or **Sort** > **Descend** from the shortcut menu. To remove the sort condition, click **Sort** > **No Sort** from the shortcut menu.

Sorting data in tables

You can sort the records or groups at a certain group level in a table.

- **Sorting records**: Changing the order of records in the whole table, or in each group if there are.
- **Sorting groups at a group level**: Changing the order of groups at the specified group level, that is, the groups will be sorted by value of the group field.

You can take the following ways to sort data in tables.

Using shortcut menu

1. Right-click on any value of a detail field or group field by which to sort the data in the table.
2. Choose the command **Sort** > **Ascend** or **Sort** > **Descend** from the shortcut menu.
3. If a detail field value is clicked on, the sorting will affect the order of detail records in the table; if it is a group field value, the order of groups in the group level represented by the group field will be rearranged.

To remove the sort condition on a field, click **Sort** > **No Sort** from the shortcut menu.

Using labels

You can use labels to control the sorting order for certain fields in tables. This feature needs to be enabled at report design time.

1. In JReport Designer, select a label in a table in a web report, and then set the Sortable property of the label to true.
2. Set the field by which you want to sort records as the value of the label's Bind Column property.
3. Save the report and publish it to JReport Server.
4. Run the report in Web Report Studio. You can see a sort button displayed beside the label.
5. To sort the table based on the specified column values ascendingly, click the upward triangle; to sort the values descendingly, click the downward triangle.

   After applying a sort order, the corresponding triangle will be highlighted. To remove the sort order, click the highlighted triangle.

Using server profile setting

You can also configure the server profile to enable sorting records in tables by clicking the sort button on table column headers.

1. Open the Profile > Customize Profile > Common tab.
2. Check the option **Sort on Column Headers** to enable clicking the sort button on column headers to sort table records (by default, this option is checked).
3. Check **Show Sort/Filter Status on Column Headers** if you want the the sort button to be always displayed on the table column headers with the current sort status. If this option is unchecked, each time you want to get the sort button for a column, you need to put the mouse pointer on its header to make the button shown.
4. Run a web report containing a table in Web Report Studio and you can click on the table column headers to sort data.

   To sort the table based on any column values ascendingly, click the upward triangle on the column; to sort the values descendingly, click the downward triangle. If the column is a group column, the order of groups in this group level will be rearranged.

   When a sort order is applied, the corresponding triangle will be highlighted. To remove the sort order, click the highlighted triangle.

Notes:

- When sorting the data in a table by a field and then by another field, the later sort condition is the major order, and the former condition the minor order, that is to say, the records in the table will be sorted by the later sort condition first, then by the former sort condition.
- The Sort on Column Headers feature does not work on summary columns that contain multiple summary fields.
- When a table is designed with the Sort by Label feature in JReport Designer, if the Sort on Column Headers option is also enabled on the server profile, the settings defined in JReport Designer have the higher priority for the specified columns.

Applying CSS styles

CSS styles can be applied to web reports formatting their appearance and characteristics. You can create and set up your own CSS styles in JReport Designer or any other CSS editor. When you publish your catalogs to JReport Server, you can include these custom styles with the catalogs.

Styles can be applied to a table, crosstab, or chart individually, or to multiple components or the whole report at a time. The last style always takes effect when more than one style is applied to the same object.

Applying a style to a report

When a style is applied to the whole report, all components in the report will take a uniform appearance. You can apply a style in one of the following ways:

- Specify a style in the Style page of the Web Report Wizard.
- In Web Report Studio, click the blank area outside of the report field, then select a style from the Report Style list on the Edit menu.

Applying a style to a component

You can apply a style to a table, crosstab, or chart by selecting the component in Web Report Studio and then doing one of the following:

- Select a style from the Report Style list on the Edit menu.
- Right-click and then select a style from the Apply Style list on the shortcut menu.

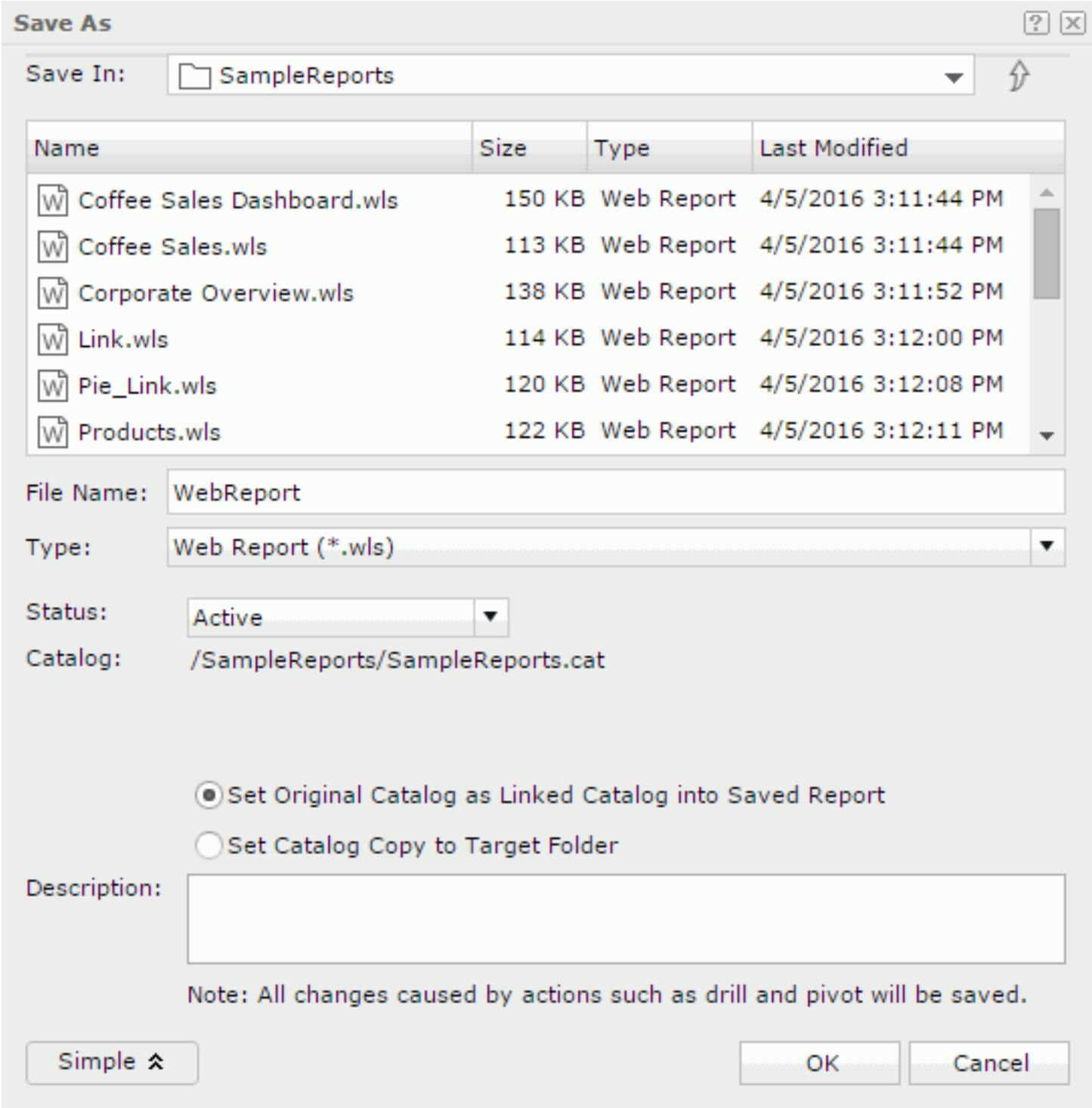
Applying a style to multiple components

You can apply a style to multiple tables, crosstabs, and charts by selecting the components and then selecting a style from the Report Style list on the Edit menu.

Saving the report

To save the changes you made to the current report, click **Menu** > **File** > **Save** (or the **Save** button on the Standard toolbar).

If the report is newly created and has not yet been saved, the Save As dialog appears.



1. In the Save in section, browse to the folder where you want to save the web report in the server resource tree. The folder may be Public Reports or My Reports. You can use the button to return to the parent folder.

   The resource table shows the resources in the current directory. Click the column names to change the order of the report in the table list if required.

2. In the File Name box, enter the name of the report or use the default name. The default file type is web report.

3. Click the **Advanced** button to set the advanced settings for the report if required.

   a. From the Status drop-down list, specify a status for the report.

   b. The catalog that the report uses is shown.

      Specify the relationship between the saved report and the catalog used to run it:

      - **Set Original Catalog as Linked Catalog into Saved Report**
        If checked, the saved report will be linked with the catalog and will run with the catalog no matter whether the two are in the same directory. If later the catalog is updated, the saved report will run with the latest version of the catalog.

      - **Set Catalog Copy to Target Folder**
        If checked, the catalog will be copied to the directory where the report is saved and the saved report will run with the copied catalog.

   c. Optionally, input comments in the Description box as a description for the report.

4. Click **Save** to save the report.

To save a copy of a report, click **Menu** > **File** > **Save As** (or the **Save As** button on the Standard toolbar) to show the Save As dialog, and then do as above.

To find a newly saved web report version, browse to select the row that the web report is in on the JReport Console > Resources page, click **Tools** > **Version** on the task bar, and then click the **Web Report Versions** tab.

Note: You will not be able to save the report to some locations if you do not have the required permissions. You need to have Write access to the directory.

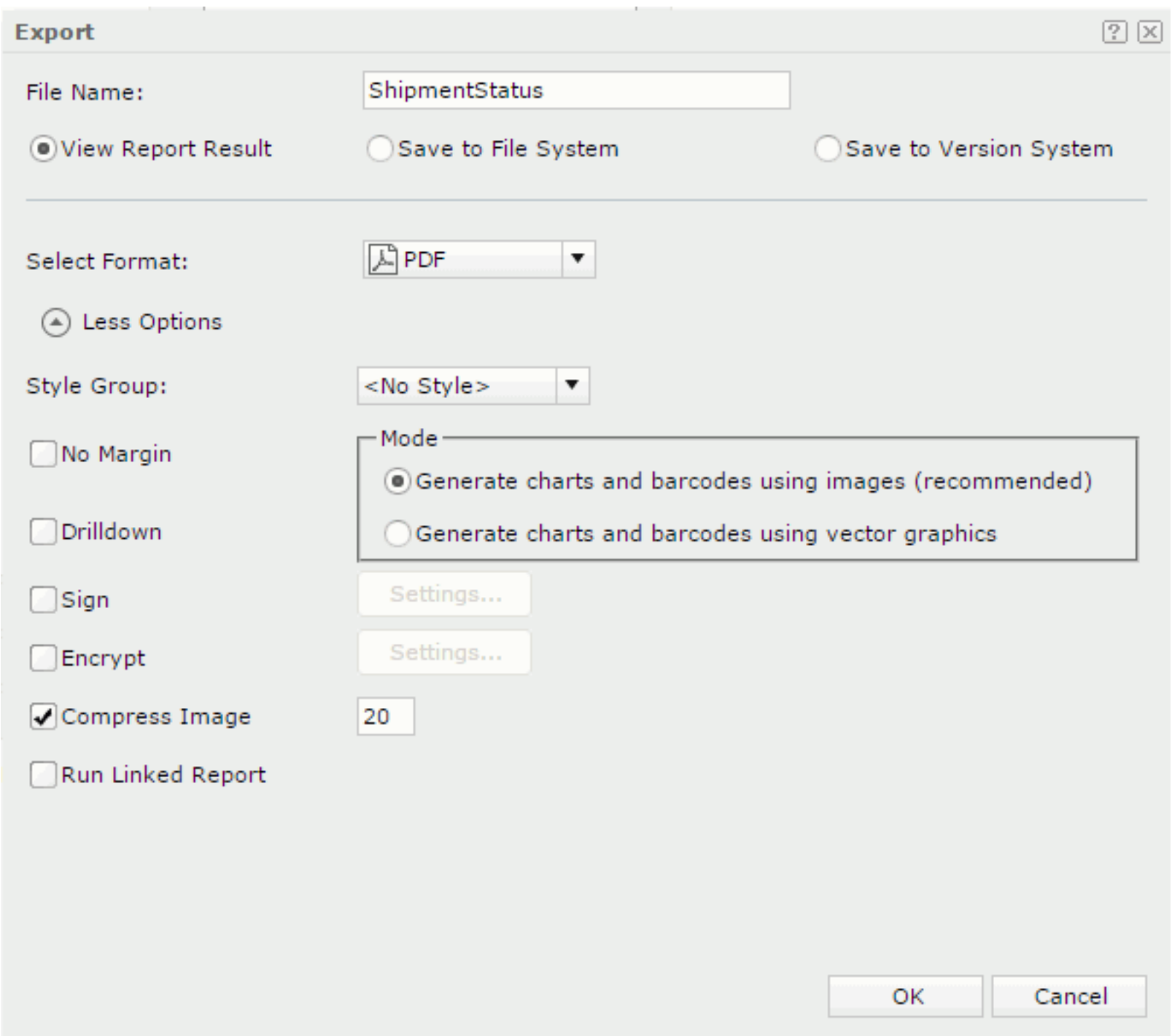
Exporting/Printing the report result

When you finish editing a web report, you may want to export it to other formats or have it printed.

Exporting the report result

You can export the report result as a result version or as a local file in these formats: HTML, PDF, Text, Excel, RTF, XML, and PostScript.

1. Click **Menu** > **File** > **Export**, or click the **Export** button on the Standard toolbar. The Export dialog appears.



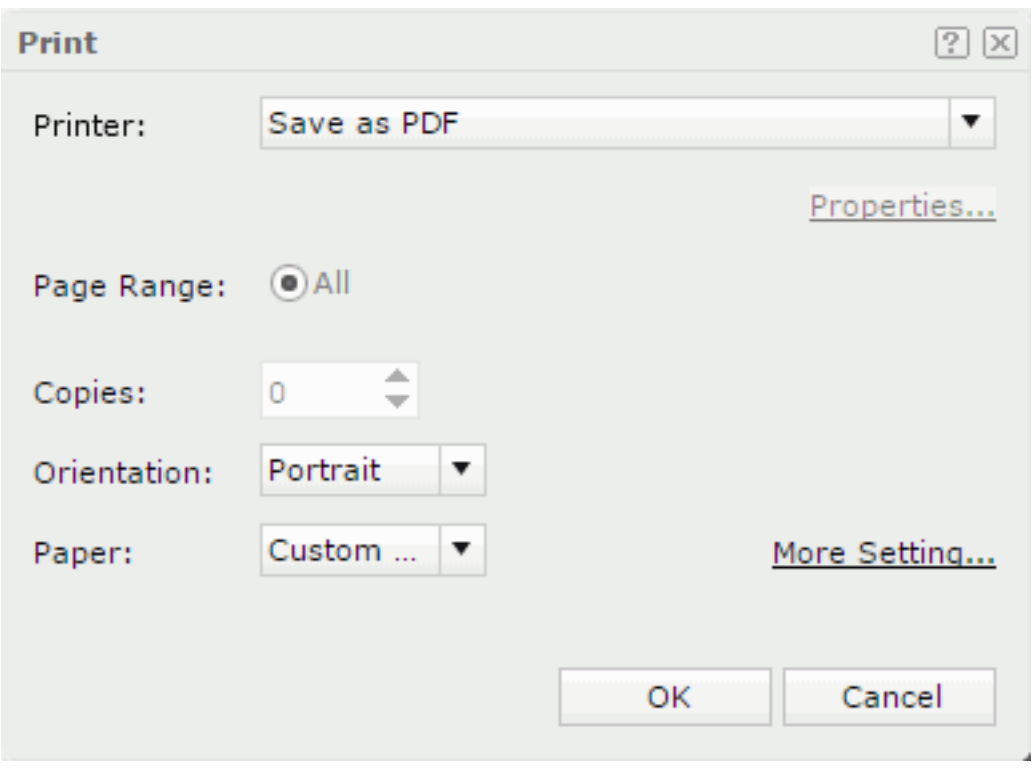
2. In the File Name field, specify the name of the exported result file.
3. Specify the destination of the result:
   - **View Report Result**: The result will be directly opened in the web browser if the format is supported by a plug-in of the web browser; otherwise it will prompt you to save the result file.
   - **Save to File System**: The web browser will prompt you to save the result file to a specified folder. If selected, you need to provide a name for the result file in the File Name field.
   - **Save to Version System**: The result will be saved as a result version in JReport Server's versioning system.
4. From the Select Format drop-down list, select the format in which to export the result.
5. Expand **More Options** to set the options for the selected format.
6. Click **OK** to confirm.

Printing the report result

You can print the report result by using a selected printer.

To print the result of a web report:

1. Click **Menu** > **File** > **Print**, or click the **Print** button on the Standard toolbar. The Print dialog appears.



2. From the Printer drop-down list, select the printer using which to print the report result. All the printers the web browser can access will be listed here. If you want to download the result to a PDF or HTML file and then use your local printer to print the file, select **Save as PDF** or **Save as HTML**.
3. If a printer is selected, click the **Properties** link to specify the printing properties in the Printer Properties dialog, then specify the number of copies you want to print accordingly.
4. Specify the page orientation and paper type for the printed result. If necessary, click the **More Settings** link to set more page properties such as paper size and margins in the Page Setup dialog.
5. Click **OK** to start printing. If Save as PDF or Save as HTML is selected in the Printer drop-down list, the result file of the web report is then opened in an associated program with which you can print the result to a printer.

JDashboard

JDashboard is a new way of information delivery, using a user portal user interface rather than a web report or page report. Users can create, edit and browse dashboards from the JReport Console using JDashboard. Library components are the basic members in dashboards for presenting data via intuitive components such as charts, crosstabs, tables, and geographic maps. Library components are created using JReport Designer, and then published to the component library on JReport Server for use when creating or modifying dashboards. With pre-built library components, users can freely choose the objects they want to display in the dashboard, without having to know how these objects were created, what data sources to use, what styles to set, and so on. A dashboard can hold multiple library components so that when browsing the dashboard users are able to see multiple data aspects. Within a dashboard, library components are able to communicate with each other via the message mechanism. This allows actions such as common filters to be applied to all the components of a dashboard even when coming from different data sources.

Since predefined containers are used for holding objects, it is easy to move objects around and resize them in dashboards.

This chapter covers the following topics:

- JDashboard basic concepts
- JDashboard window elements
- Creating dashboards
- Inserting components
- General operations in JDashboard
- Manipulating data components
- Filtering component data
- Specifying parameter values
- Saving dashboards
- Exporting library components
- Printing library components
- Setting JDashboard as the server home page
- Running dashboards via URL
- Running and editing reports in JDashboard
- Opening Visual Analysis in JDashboard

Notes:

- A JDashboard license is required in order to use JDashboard. If you do not have the license, contact your Jinfonet Software account manager to obtain it.
- JDashboard cannot run on Internet Explorer 8.

JDashboard basic concepts

The following shows the main JDashboard concepts:

Library components

Library components are used to build dashboards. They are able to present data via intuitive components such as charts, crosstabs, tables, and geographic maps. Library components can be created and edited using JReport Designer, and then published to JReport Server for use in dashboards. Library components can also come from report components. JReport Designer provides a way to save a data component in a web report to a library component. Or when a dashboard user inserts a web report component into a dashboard, the report component will be converted to a library component and the user can save the library component. Library components use .lc as the file suffix.

Component library

Component library contains one Public Components folder and a My Components folder for each specific dashboard user.

Public Components and My Components are two built-in folders in the server resource tree root for storing library components. The Public Components folder contains public components available to everyone. The My Components folder holds personal components for each dashboard user.

Dashboard

A workspace window that can contain any number of library components.

- **Components from library**
  When inserting a library component from the component library into a dashboard, you are not copying the component from the library, but instead referencing it from the library, in this sense, the changes to the component in the library will be reflected in all of the dashboards referencing the component. The contents of library references in dashboards cannot be edited since they are referenced resources.

- **Report components**
  Data components such as tables, charts, crosstabs, and geographic maps in reports can be inserted into dashboards as library components.

- **Objects from the Toolbox**
  In addition to library components users can select objects from the Toolbox such as labels, images, special fields, sliders, filter controls, third-party objects, and HTML components.

Message

The information that one library component can send to another. The sending and receiving of messages between library components are defined in JReport Designer.

JDashboard window elements

JDashboard is opened in a web browser. The full-featured JDashboard window is composed of three sections: the dashboard title bar at the top, the side bar on the left providing options for working with dashboards, and the dashboard editing area where you navigate and modify dashboards. Since JDashboard can also be used to access Visual Analysis, if your JReport Server is enabled with Visual Analysis, the elements on the JDashboard window have slight differences.

The ways of accessing JDashboard

Here are two ways of accessing the JDashboard:

By creating a new dashboard

On the JReport Console > Resources page, click **New** > **Dashboard**. The JDashboard window will be launched with a blank dashboard. This way enters the edit mode.

By running an existing dashboard

On the JReport Console > Resources page, browse to the target dashboard and choose the appropriate way to enter the desired view or edit mode.

View mode and edit mode

JDashboard has two working modes: the edit mode presents full functionalities and the view mode just serves for viewing purpose without any editing abilities. After JDashboard is open, you cannot switch one mode to another. If you need to do this, reopen JDashboard using the appropriate way.

To enter the view mode:

- Browse to locate an existing dashboard on the JReport Console page, then click the **Run** button on the floating toolbar, or click **Run** > **Run** on the standard toolbar.

To enter the edit mode:

- On the JReport Console page, click **New** > **Dashboard**.
- Browse to locate an existing dashboard on the JReport Console page, then click the **Edit** button on the floating toolbar, or click **Run** > **Edit** on the standard toolbar.

You can also directly click the name of a dashboard to enter the view mode if the dashboard is in the My Reports folder, or either of the two modes if it is in the Public Reports folder, depending on the permission you are granted on the dashboard:

- When you have the Execute permission on the dashboard, you will be directed to the view mode.
- When you have the Edit permission on the dashboard but do not have the Execute permission, you will enter the edit mode.
- When you have both the Execute and Edit permissions on the dashboard, you will enter the view mode.

The following introduces the full UI elements based on the edit mode.

Dashboard title bar

The dashboard title bar at the top contains tabs labeling the names of the open dashboards.

Dashboard name tabs

Each tab represents an open dashboard and the dashboard name is shown on the tab.

The following are operations on the tabs:

- Click a tab to activate the corresponding dashboard.
- Rename a tab. Double-click a tab name to enter the editing mode. After typing a new name, press Enter or click outside of the input field to save the name.
- Move a tab. Drag a tab and drop it beside a different tab so as to change the tab order.
- Click **x** beside a dashboard name to close the dashboard.

Adds a new blank dashboard in the current web browser.

If Visual Analysis is enabled, you can leave the cursor on the button for one second and a drop-down menu will be displayed containing the following options:

- **New Dashboard**
  Creates a new blank dashboard as a new tab.
- **New Analysis**
  Opens Visual Analysis as a new tab.

English (United States)

Specifies the language in which the dashboard will be displayed. Available only when Enable NLS is checked in the Profile > Customize Server Preferences > Advanced panel on the JReport Console page.

Dashboard editing area

Dashboard view has header and body.

The dashboard header can contain labels, images, and special fields.

The dashboard body can contain report components, library components, filtering tools, third-party objects, and HTML components. You can insert the same library component repeatedly to the same dashboard body.

Library components inserted to dashboards are references of the library component in component library. The changes to a library component in the library will be reflected in all of the dashboards referencing the library component, such as removal of library components, version updated, permission changed.

Side bar

The options available on the side bar are determined by the feature profile that is selected as the default profile in the Profile > Configure Profile > JDashboard > Features tab and the property setting on the Profile > Configure Profile > JDashboard > Properties tab (profile has the higher priority). The default JDashboard profile provides full options.

The side bar on the left contains these buttons:

Shows or hides the Resources panel which includes these branches:

- **Component Library** lists the library components created in and published from JReport Designer. You can select a library component and drag it into the dashboard body.
- **Reports** lists page reports and web reports. You can open a report from JDashboard or add report data components into your dashboards.
- **Toolbox** lists the objects that can be inserted in your dashboards such as labels, images, special fields, filtering tools, third-party objects, and HTML components.

You can use the quick search toolbar to search for any desired resources in a fast and convenient way (to display the toolbar, click the **Search** button at the upper right corner of the panel).

- **Text box**
  The quick search toolbar treats resources as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the resources.
- **X**
  Closes the quick search toolbar.
-
  Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
-
  Highlights the previous matched text.
-
  Highlights the next matched text.

New

Creates a new blank dashboard with a new tab added.

If Visual Analysis is enabled, clicking the button will bring out a drop-down menu that contains the following options:

- **New Dashboard**
  Creates a new blank dashboard as a new tab.
- **New Analysis**
  Opens Visual Analysis as a new tab.

Open

Displays the Open Dashboard dialog for you to specify a dashboard to open, or the Open Document dialog where you can select a dashboard or a Visual Analysis template to open. Which dialog will be displayed depends on whether Visual Analysis is enabled.

Save

Saves the changes made to the current dashboard.

Refresh

Refreshes the current dashboard.

Enter Parameter Values

Opens the Enter Parameter Values dialog which lists all the parameters used in the dashboard for specifying their values.

Clear Filters

Removes all the filters from the current dashboard including those generated via sliders, filter controls, messages, drilling and going actions and those designed using web browsers such as Page Report Studio and Web Report Studio, except query filters and others designed and taking effect in JReport Designer.

Arrange

Automatically arranges the library components in the current dashboard to be tidy. It is an instant action.

Export

Displays the Export dialog for exporting the library components in the current dashboard.

Print

Displays the Print dialog for printing the library components in the current dashboard.

Options

Displays the following options:

- **New**
  Creates a new blank dashboard.

  If Visual Analysis is enabled, the New option contains a submenu:
  - **Dashboard**
    Creates a new blank dashboard as a new tab.
  - **Analysis**
    Opens Visual Analysis as a new tab.
- **Open**
  Displays the Open Dashboard dialog for you to specify a dashboard to open, or the Open Document dialog where you can select a dashboard or a Visual Analysis template to open. Which dialog will be displayed depends on whether Visual Analysis is enabled.
- **Save**
  Saves the changes made to the current dashboard.
- **Save As**
  Saves the dashboard with a different name or to a new location.
- **Export**
  Displays the Export dialog for exporting library components in the current dashboard.
- **Print**
  Displays the Print dialog for printing library components in the current dashboard.
- **Clear Filters**
  Removes all the filters from the current dashboard including those generated via sliders, filter controls, messages, drilling and going actions and those designed using web browsers such as Page Report Studio and Web Report Studio, except query filters and others designed and taking effect in JReport Designer.
- **Share Parameter**
  Opens the Share Parameters Setting dialog for sharing parameters between library components.
- **Arrange**
  Automatically arranges the library components in the current dashboard to be tidy. It is an instant action.
- **Auto Arrange**
  Automatically arranges the library components in the current dashboard to be tidy once the layout requires arrangement. It is a status setting. After selecting it, this option will be kept checked and the Arrange option will be disabled until you unselect Auto Arrange. The status will be saved with the dashboard.
- **Set as Server Home**
  Sets the current JDashboard status as the home page after logging onto JReport Server. For details, see Setting JDashboard as the server home page.
- **Language**
  Allows you to specify the language in which the dashboard will be displayed. Available only when Enable NLS is checked in the Profile > Customize Server Preferences > Advanced panel on the JReport Console page.
- **Component Title Bar**
  Customizes the way of showing component title bar and the icons on it.
- **Themes**
  Opens the Themes dialog for selecting a theme to apply to the current dashboard.
- **Show / Hide Dashboard Header**
  Changes the current status of the dashboard header from being shown to hidden or from being hidden to shown.
- **Help**
  Displays the JDashboard help documents.
- **Exit**
  Exits JDashboard.

Component options menu

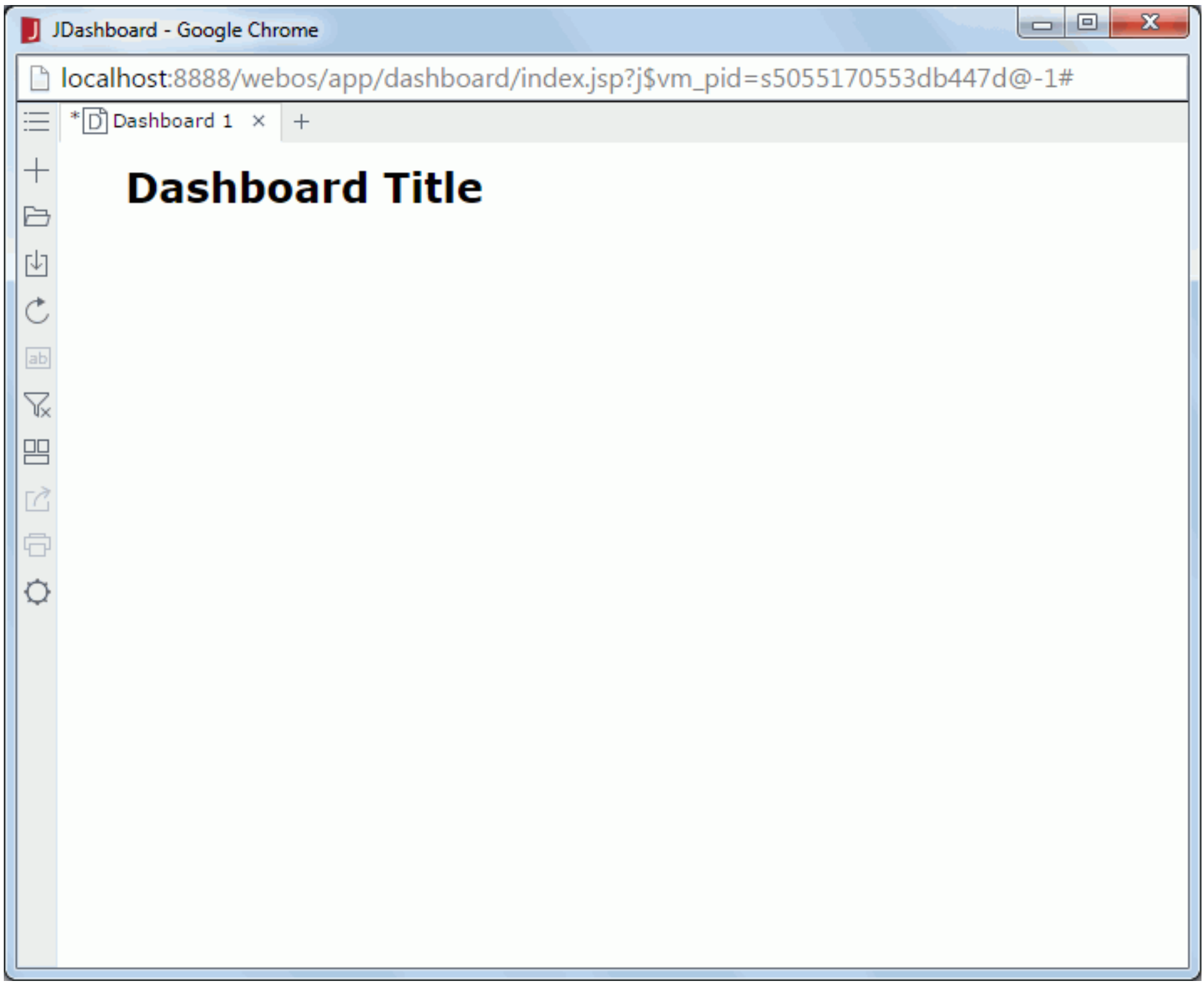
The component options menu is available when you click "" on the title bar of a library component. The options available on the menu are determined by the feature profile that is selected as the default profile in the Profile > Configure Profile > JDashboard > Features tab and the property setting on the Profile > Configure Profile > JDashboard > Properties tab (profile has the higher priority). The default JDashboard profile provides full options.

- **Analyze**
  Opens the Visual Analysis component in a new JDashboard tab or browser window.
- **Edit Setting**
  Edits the component setting in the corresponding edit dialog.
- **Export**
  Exports the library component.
- **Delete**
  Deletes the library component from the current dashboard.
- **Refresh**
  Customizes the auto refresh action for the library component.
- **About**
  Displays the component information such as its ID, the author and his/her e-mail address, and the description about the component.

Creating dashboards

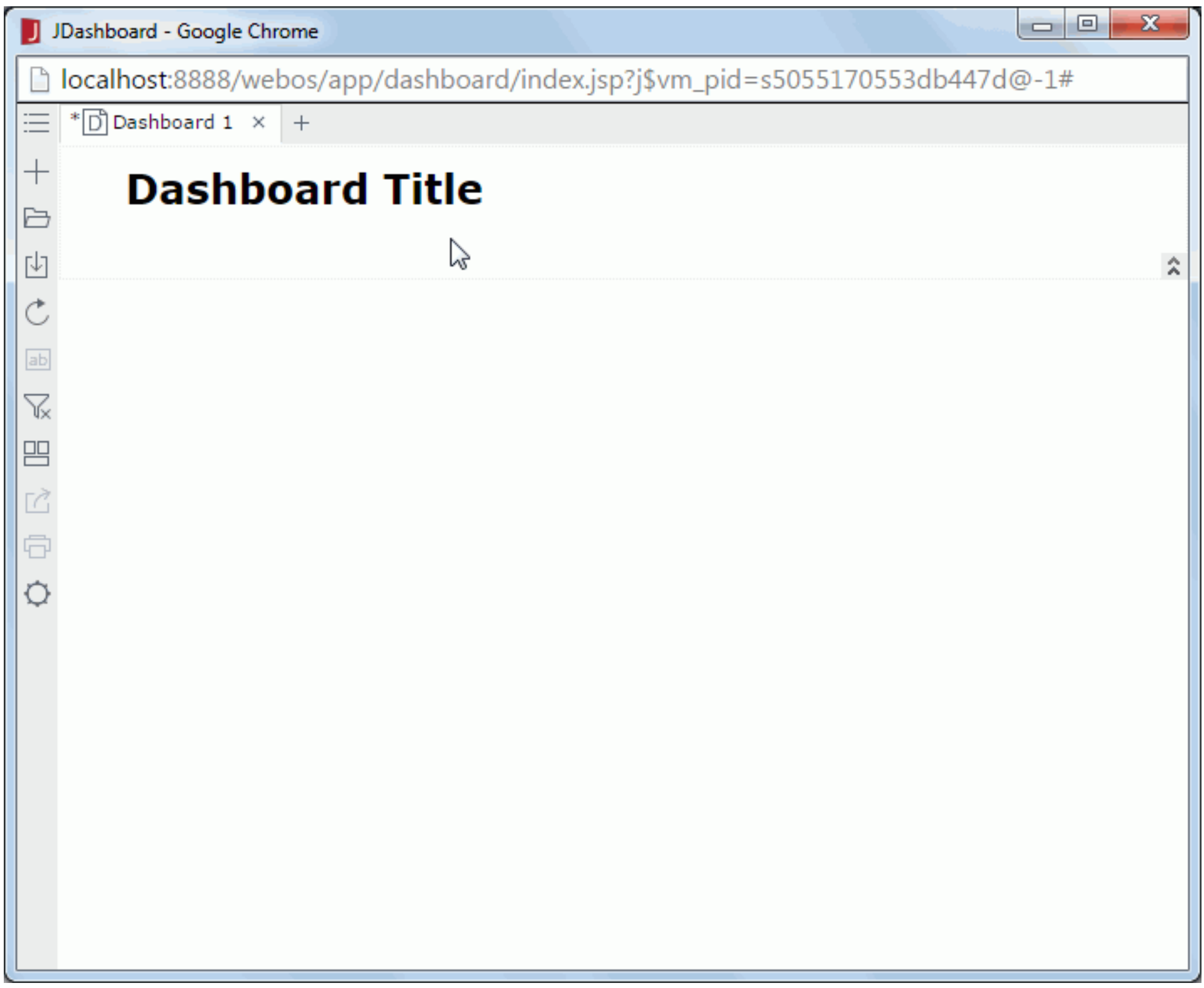
To create a dashboard:

1. On the JReport Console > Resources page, click **New** > **Dashboard**. A blank dashboard will be created.



2. The upper section with the text "Dashboard Title" of the editing area is the dashboard header, where you can insert labels, images, and special fields. The dashboard body is the section below the header. In the body you can insert report components, library components, filtering tools, third-party objects, and HTML components. For details see Inserting components.

3. When the mouse hovers on the header section, the header is outlined and you will see the border between the header and the body. You can resize the two sections by dragging the border line. The arrow button that appears at the bottom right of the header is used to hide the header.



After you have entered JDashboard, if you want to create more dashboards in the same web browser, see here.

Inserting components

You can insert library components and report data components as well as labels, images, special fields, filtering tools, third-party objects, and HTML components into dashboards via the Resources panel. To access the panel, click **Show Resources** on the side bar.

The following shows inserting specific components in detail.

Inserting library component references

When inserting a library component from component library into a dashboard, you are not copying the component from the library, but instead referencing it from the library, in this sense, the changes to the component in the library will be reflected in all of the dashboards referencing the component. The contents of library references in dashboards cannot be edited since they are referenced resources.

To reference a library component into the dashboard body:

1. Click **Show Resources** on the side bar to display the Resources panel.
2. Expand the **Component Library** node, browse to find the library component you want to insert, then drag it to the destination in the dashboard body.

Inserting report data components

Data components such as tables, crosstabs, charts, and geographic maps in existing page reports or web reports can be directly inserted into dashboards after being converted into library components automatically.

For reports that use business views as data sources, all data components can be converted to library components successfully.

However, page reports may use queries as data sources other than business views, therefore, only when the queries have corresponding business views, can the components be converted to library components and used in dashboards. So if you would like your page report components to be added in dashboards, you need to make sure the components are created on business views or if on queries a business view is created for each of the queries.

Currently library components do not support some features of page report components, after the latter are inserted into dashboards, those features will be removed. This may result in that the data components in dashboards looks different from they are in page reports. For features that are not supported in JDashboard, they will either be ignored, removed, or applied with the default values.

The following table lists how JDashboard deals with the unsupported page report features:

In Page Report Components	In Library Components
Display types like Barcode, Check box, and so on	Ignored
Special fields	Removed
Dynamic resources	Changed to constant resources
Master/Detail reports	Ignored
Subreports	Removed
Nested data components that is one contains another	The ownership is removed and the involved data components are regarded as individual components.
Definition properties	Ignored
Formula-controlled properties	Default values are applied.
Other components	Removed

To insert a report data component into the dashboard body:

1. Click **Show Resources** on the side bar to display the Resources panel.
2. From the Reports node, expand the report that contains the wanted data component and drag the component into the dashboard body.
3. If the report component uses parameters, the default values will be applied. To change the parameter values, after the component is loaded, hover the mouse over its title bar, then click "" that appears on the title bar and select **Edit Setting** from the drop-down list to display the configuration panel. Change the parameter values in the panel and then click **OK** to apply the new values.

The inserted report data component runs with the report's catalog. It will not be able to run if the catalog is removed or updated. The report component does not synchronize with the other library components in the dashboard via messages and is not controlled by the runtime sliders or filters.

Inserting Visual Analysis components

Visual Analysis components after being saved into the server resource tree can be inserted into dashboards. This feature requires a license for Visual Analysis.

To insert a Visual Analysis component into the dashboard body:

1. Click **Show Resources** on the side bar to display the Resources panel.
2. From the Reports node, browse to the target Visual Analysis template (.va) and you will find a VCTOBject under it. Drag the VCTOBject into the dashboard body.

The inserted Visual Analysis component will be wrapped into a runtime library component and its data presentation area and legend area without legend icons will be displayed, just for view. However if no field binds with any legend, the legend area will not be shown. The default title is the server resource name of the parent Visual Analysis template without the suffix .va.

If the Visual Analysis component uses parameters, you can change the parameter values as you want. To do this, hover the mouse over its title bar, then click "" that appears on the title bar and select **Edit Setting** from the drop-down menu. In the configuration panel, edit the parameter values.

You can enable the Visual Analysis component to receive the built-in 0001 - Filter messages sent out by other library components in the dashboard. To do this, click "" on the component title bar and then select **Enable Receiving Filter Message** from the drop-down menu.

Dashboard filters can work on Visual Analysis components.

Visual Analysis components cannot be exported or printed, therefore when exporting or printing a dashboard, they will not be listed and the cells containing them will be left blank.

Inserting a label

Labels can be inserted in the dashboard header. To do this, click **Show Resources** on the side bar to display the Resources panel, then from the Toolbox node, drag **Label** to the destination in the dashboard header. Double-click the label and edit the text as required. If you want to further format the label, hover the cursor on the label and click the button in the appearing dotted rectangle to open the Edit Label dialog, in which you can edit the properties of the label such as font, size, color, and so on.

Inserting a dashboard title

A dashboard title is a special label. It can be inserted in the dashboard header. To do this, click **Show Resources** on the side bar to display the Resources panel, then from the Toolbox node, drag **Dashboard Title** to the destination in the dashboard header. Double-click the title and edit the text as required. If you want to further format the title, hover the cursor on the title and click the button in the appearing dotted rectangle to open the Edit Label dialog, in which you can edit the properties of the title such as font, size, color, and so on.

Inserting an image

Images can be inserted in the dashboard header.

1. Click **Show Resources** on the side bar to display the Resources panel.
2. From the Toolbox node, drag **Image** to the destination in the dashboard header. The Insert Image dialog appears.

3. Specify the image you want to insert.
   - To use an image in the local file system, select **Local File**, then click **Browse** to find the image.
   - To use an image on a website, select **Web URL**, then input the image URL or paste the URL in the File URL text field.
   - To use an image in the image library of JDashboard, select **Library**, then select the image in the Pictures box.
4. Click **OK** to insert the image.

Once an image is inserted in the dashboard header, if you want to replace it with another one, hover the cursor on the image and click the button in the appearing dotted rectangle. In the Edit Image dialog, select another image to use.

Inserting a special field

You can insert these types of special fields in the dashboard header:

- **User Name**
  The user name with which you log onto JReport Server.
- **Modified Date**
  The date when the dashboard was last modified.
- **Modified Time**
  The time when the dashboard was last modified.
- **Print Date**
  The date to run the dashboard.
- **Print Time**
  The time to run the dashboard.

To do this:

1. Click **Show Resources** on the side bar to display the Resources panel.
2. From the Toolbox node, drag **Special Field** to the destination in the dashboard header. The Insert Special Field dialog appears.

3. Choose the desired special field and click **OK** to insert it into the header.
4. To format the special field, hover the cursor on the special field and click the button in the appearing dotted rectangle.
5. In the Edit Special Field dialog, edit the properties of the special field such as font, size, color, and so on.

Inserting a slider

Sliders can be inserted in the dashboard body. They are used to filter component data. For details, see Using sliders.

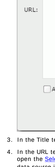
Inserting a filter control

Filter controls can be inserted in the dashboard body. They are used to filter component data. For details, see Using filter controls.

Inserting a third party gadget via URL

A web page can be inserted in dashboards. All you need to do is give its URL. Note that some websites such as http://www.google.com do not allow Gadgets to load them.

1. Click **Show Resources** on the side bar to display the Resources panel.
2. From the Toolbox node, drag **URL Frame** into the dashboard body. The Insert URL Frame dialog appears.



3. In the Title text field, give a title for the frame that will display the contents of the web page.
4. In the URL text box, type in the URL of the web page. If needed, click the **Add Parameter** button to open the Select Parameter dialog to insert a parameter to the URL to compose a dynamic URL (if no data source is specified, the Select Data Source dialog will be displayed for you to select the data source that contains the parameter you want firstly).

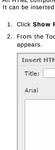
   You should provide a complete URL address. A URL without "http://", for example www.jinfonet.com, will not be automatically added "http://" since it is regarded a relative path, which may lead to that the URL cannot be opened in some browsers.
5. If you would like the specified web page to refresh periodically, select **Auto refresh**, then specify the time interval at which to refresh it.
6. Click **OK**. The specified web page will be inserted into the dashboard. You can then view the web page from JDashboard. If parameters are used in the URL, you can click on the side bar to specify the parameter values as you want, then you can get different web pages based on different parameter values.

You can further modify the web page frames inserted into a dashboard if you want. To do this, hover the mouse over the title bar of a web page frame, then click "" that appears on the title bar and select **Edit Setting** from the drop-down list. In the Edit URL Frame dialog, edit the setting as required.

Inserting an HTML component

An HTML component allows for typing text, comments, and messages using a simple-featured text editor. It can be inserted in the dashboard body.

1. Click **Show Resources** on the side bar to display the Resources panel.
2. From the Toolbox node, drag **HTML** to the destination in the dashboard body. The Insert HTML dialog appears.



3. Specify a title for the HTML component.
4. In the text box, type text directly. You can make use of the buttons above the text box to format the text such as font face, size, style, color, and alignment, insert images into the HTML component and create hyperlinks on the text and images in the HTML component.
5. Click **OK** to insert the HTML component.

You can further modify the HTML components inserted into a dashboard if you want. To do this, hover the mouse over the title bar of an HTML component, then click "" that appears on the title bar and select **Edit Setting** from the drop-down list. In the Edit HTML dialog, edit the setting as required.

General operations in JDashboard

This section provides a general view of the operations you can perform in JDashboard.

Operations on dashboards

- **Creating a new blank dashboard in the current web browser**
  The ways to create a dashboard differ depending on whether your JReport Server is enabled with Visual Analysis.
  - When Visual Analysis is disabled, choose a method from the following:
    - Click **+** beside the rightmost tab on the dashboard title bar.
    - Click the **New** button on the side bar.
    - Click the **Options** button on the side bar and select **New** from the option list.
  - When Visual Analysis is enabled, choose a method from the following:
    - Click **+** beside the rightmost tab on the dashboard title bar.
    - On the dashboard title bar, leave the mouse cursor on the **+** button beside the rightmost tab for one second until a drop-down menu displays, then select **New Dashboard** from the menu.
    - On the side bar, click the **New** button and then select **New Dashboard**.
    - On the side bar, click the **Options** button and then select **New** > **Dashboard**.

  A blank dashboard will be created in the browser. You can then add components to customize the dashboard.
- **Opening another dashboard in the current web browser**

  Click the **Open** button on the side bar, or click the **Options** button and then select **Open** from the option list. In the appearing dialog, browse to the desired folder, select the dashboard you want to open and then click **OK**, or simply double-click the dashboard to open it.
- **Showing, hiding and resizing the dashboard header**

  To show or hide the dashboard header, click the **Options** button on the side bar and select **Show/Hide Dashboard Header** from the option list.

  Even though the dashboard header is open, its border is unseen by default. However, when you drag objects into the dashboard editing area, you will see the area is divided into two sections with the activated section highlighted in grey: the upper section is the header and the lower the body. Or you can simply press **Ctrl** on the keyboard to make the border between the header and body shown. Then drag the border line vertically to adjust the size of the header and body.
- **Removing all filters from a dashboard**
  JDashboard provides the ability to remove all the filters from the current dashboard at a time including those generated via sliders, filter controls, messages, drilling and going actions and those designed using web browsers such as Page Report Studio and Web Report Studio, except query filters and others designed and taking effect in JReport Designer. To do this, click the **Clear Filters** button on the side bar, or click the **Options** button and then select **Clear Filters** from the option list.
- **Refreshing the current dashboard**

  Click the **Refresh** button on the side bar to refresh the data of the current dashboard.
- **Applying a theme to the current dashboard**

  Click the **Options** button on the side bar and select **Themes** from the option list. In the Themes dialog, select a theme from the left panel and then click **OK**.
- **Closing a dashboard**
  For the current dashboard, click **X** on its name tab to close it. For the other open dashboards, hover the cursor on a name tab until X appears, then click **X** to close that dashboard.
- **Exiting JDashboard**

  If you want to exit JDashboard and release the resources, click the **Options** button on the side bar and select **Exit** from the option list. Do not use the close button on the browser window as that may not release the resources used by JDashboard.
- **Asking for help**

  At any time, you can click the **Options** button on the side bar and select **Help** from the option list to access the JDashboard help documents. Furthermore, you can click in any dialog to show the help document about the dialog.

Operations on objects in the dashboard header

- **Editing an object**
  Hover the cursor on the object and then click that appears around the object. A dialog will be displayed for you to edit the object. For a label or the dashboard title, you can edit the text related properties like font face, font color, and so on. For a special field, you can change it to another special field and edit the text related properties. For an image, you can select another image to replace the current one.
- **Moving an object**
  Place the cursor on the object until it becomes a four-arrow icon, then drag to the desired position.
- **Resizing an object**
  Place the cursor on the object until an orange rectangle appears, next move the cursor on the right border, bottom border, or the bottom right corner until the cursor becomes a two-arrow icon, then drag to the desired position.
- **Deleting an object**
  Hover the cursor on the object and then click that appears around the object.

Operations on components in the dashboard body

- **Moving a component**
  Place the cursor on the component title bar until it becomes a four-arrow icon, then drag to the desired position.
- **Resizing a component**
  Place the cursor on the component's right border, bottom border, or the bottom right corner until the cursor becomes a two-arrow icon, then drag to the desired position.
- **Editing a component**
  Hover the mouse over the title bar of the component, then click that appears on the title bar and select **Edit Setting** from the drop-down list. In the corresponding edit dialog, edit the setting as required.
- **Exporting a component**
  Hover the mouse over the title bar of the component, then click that appears on the title bar and select **Export** from the drop-down list. In the Export dialog, choose the format to which you want to export the library component, then click **OK**.
- **Navigating component data via scroll bar**
  For tables, crosstabs and charts, you can use the scroll bar to navigate their data if the container cannot display all data of the component.
- **Turning component pages**
  If a table or a crosstab contains more than one page, a page navigation bar specific for the component will be available right below the component. You can use the navigation bar to view the desired pages:
  - Click a number to go to that page.
  - Input a number in the text field.
  - Click **Prev** to go to the previous page.
  - Click **Next** to go to the next page.
- **Customizing the auto refresh action**
  For a library component that uses only one business view as the data source, if its Auto Refresh feature is enabled, you can hover the mouse over its title bar, then click that appears on the title bar and select Refresh from the option list to adjust the refresh interval or pause the auto refresh action with the Refresh Interval dialog.
  - To change the refresh interval, click the drop-down list to select the new interval. Currently there are only 3 values available, which are ordinal multiples (starting from 2) of the default interval value defined in JReport Designer.
  - To make the auto refresh action paused, click .
  - To resume the auto refresh action after it is stopped, click .
- **Going to links**
  Once an object in a library component has been linked to a report, a web page, or an e-mail address, you will find the mouse cursor changes into a hand icon when hovering on the object, you can then click the object to launch the linked target.

  > **Note:** If the trigger object of a link in a library component happens to be in some hierarchies (which means you can also perform the go-down action on the object by single click), you can click the trigger object to directly open the link in JDashboard only when the click priority of the link action is specified to be the highest for the library component at design time. If the priority of the go-down action is specified the highest, you can click the trigger object to open the link only when you have reached the lowest level of the hierarchies by going down on the object first.
- **Sending messages**
  If an object in a library component has been defined to send a message, by triggering the event defined for sending the message, the message will be sent out and the components defined for receiving the message will do things as defined.
- **Maximizing a component**

  Click on the component title bar and the component will take up the whole dashboard body. By clicking the button again the component will be restored to the original size.
- **Editing the title of a component**
  Double-click on the component title and it will become editable. Input the new title and click **Enter** on the keyboard to confirm the change.
- **Customizing the way of showing the component title bar and options on it**

  Click the **Options** button on the side bar and select **Component Title Bar** from the option list. In the Customize Component Title Bar dialog, specify the way of showing the component title bar and the options on it, decide whether to save the settings into your user profile via the Save into My Preferences Settings option, and then click **OK**.
- **Deleting a component**
  Hover the mouse over the title bar of the component, then click that appears on the title bar and select **Delete** from the drop-down list.
- **Making use of the configuration panel**
  Each library component can be equipped with a configuration panel. The configuration panel can be used to specify parameter values to its library component, to filter or sort the data of its library component, or to change properties of objects in its library component.

  Once a configuration panel has been defined for a library component, you can open it in JDashboard and then perform actions as defined. To do this, hover the mouse over the title bar of the library component, then click that appears on the title bar and select **Edit Setting** from the drop-down list to display the configuration panel. After specifying values in the panel, click OK to apply the values in the library component. The Cancel button is used to close the configuration panel.
- **Showing component information**
  Information about a component such as a library component's ID, the author and his/her e-mail address, and the description about the component are provided. To view the information, hover the mouse over the title bar of the component, then click that appears on the title bar and select **About** from the drop-down list. A panel will be displayed showing the information. You can click the OK button in the panel to close the panel.
- **Automatically arranging library components in the current dashboard**
  Sometimes you may drag many library components to random positions in a dashboard and would like them to look neat, there is a simple way to do this: on the side bar, click the **Arrange** button or click the **Options** button and select **Arrange** from the option list, the dashboard layout will then be adjusted immediately. Arrange is an instant action, clicking it once will arrange the layout once, and in this sense it benefits when you seldom need to reorder the layout. In the case when frequent reorder of dashboard layout is required, click the **Options** button on the side bar and click **Auto Arrange** on the option list. Then later each time a library component is resized, moved, removed, or added, the dashboard body will be rearranged automatically so that the library components will be placed tidy.

Manipulating data components

You can manipulate data components, which refer to crosstabs, tables, charts, and geographic maps, in dashboards as shown below.

Applying a style to a data component

Right-click in the component, then on the shortcut menu, select a style from the Apply Style submenu.

Going through the data of tables, crosstabs and charts

Going actions enable you to switch among data groups freely for viewing different records without having to create a new data component. They can be performed on table groups, crosstab column/row headers, and data labels on the chart category/series axis as well as the legend entry labels.

Choose the proper one to meet your needs from the following:

- **Go To**
  Jumps to a different group by replacing the current.
- **Go to by Value**
  Jumps to another group while applying the current value being clicked on as a filter condition.
- **Go Down**
  Goes from a group which is non-bottom level in a predefined hierarchy to the one-level-lower group while applying the current selected value as a filter condition.
- **Go Up**
  Jumps from a group which is non-top level in a predefined hierarchy to the one-level-higher group.

Note: You can also directly click group objects to perform the go-down action, however, if the group objects are also defined with some links, you can go down to the next level with a single click only when the click priority of the go-down action is specified to be the highest at design time. Otherwise, you have to use the Go Down command on the objects' shortcut menu to perform the action.

For example demonstration of them, refer to Going through the report data.

The go-to-by-value or go-down actions always happen along with the generation of a filter condition. The filters are displayed as "FieldName:Value" at the bottom of the library components, in a row one by one from left to right according to the time they are generated. When the row cannot hold all of the filters, two buttons are displayed at the two ends of the row for scrolling through the filters to the left and right. Each click on the button will show one hidden filter. To remove a filter condition, click **X** right to it.

Sales Crosstab

Category	State	Beijing	Hong Kong	Shanghai	Taipei	Total
Blends	Sales	$60.822	$52.605	$88.274	$29.469	$231.170
	Quantity	5.742	5.034	8.369	2.820	21.965
Bold	Sales	$14.149				$14.149
	Quantity	1.410				1.410
Exotic	Sales	$32.607	$32.974	$49.461	$58.612	$173.654
	Quantity	3.467	3.506	5.259	6.232	18.464
Flavored	Sales	$25.196		$7.362	$83.458	$116.016
	Quantity	2.533		818	9.000	12.351
Mild	Sales	$39.287			$51.540	$90.827
	Quantity	2.921			5.136	8.057
Total	Sales	$172.062	$85.579	$145.097	$223.078	$625.817
	Quantity	16.073	8.540	14.446	23.188	62.247

Region: Asia-Pacific × Country: China ×

Customizing the data format of values in tables, crosstabs and charts

To customize the data format of the values of a field in a table or crosstab, right-click on any value of the field, then select a format from the Format submenu, or input a format in the text box at the bottom of the submenu and click **Enter** on the keyboard.

You can also customize the data format of the legend entry labels and data labels on the category and series axes if they are of Number or Date/Time type. To do this, right-click on the labels and select the required format from the Format submenu, or input a format in the text box at the bottom of the submenu and click **Enter** on the keyboard.

Removing component level filters from a data component

On the shortcut menu of data components, there is an option Remove Filters which is used to remove filter conditions generated via the configuration panel and via message delivery from the data components. These two kinds of filters are referred to as component-level filters.

In dashboards you can also use sliders and filter controls to do filtering, however filters created by sliders or filter controls are not under the control of the removing component level filter action, because they are regarded as dashboard-level filters.

Manipulating a table

- **Sorting data in a table**
  - **Using shortcut menu**
    You can sort the values of a field in a table with the shortcut menu. To do this, right-click on any value of the field, then on the shortcut menu, select **Ascend** or **Descend** from the Sort submenu. If a detail field value is clicked on, the sorting will affect the order of detail records in the table; if it is a group field value, the order of groups in the group level represented by the group field will be rearranged. To remove the sort condition, select **No Sort**.
  - **Using labels**
    You can use labels to control the sorting order for certain fields in tables. This feature needs to be enabled at report design time.
    1. In JReport Designer, select a label in a table in a library component, and then set the Sortable property of the label to true.
    2. Set the field by which you want to sort records as the value of the label's Bind Column property.
    3. Save the library component and publish it to JReport Server.
    4. Create a dashboard and insert the table in it. You can see a sort button displayed beside the label.
    5. To sort the table based on the specified column values ascendingly, click the upward triangle; to sort the values descendingly, click the downward triangle.

       After applying a sort order, the corresponding triangle will be highlighted. To remove the sort order, click the highlighted triangle.
  - **Using server profile setting**
    You can also configure the server profile to enable sorting records in tables by clicking the sort button on table column headers.
    1. Open the Profile > Customize Profile > Common tab.
    2. Check the option **Sort on Column Headers** to enable clicking the sort button on column headers to sort table records (by default, this option is checked).
    3. Check **Show Sort/Filter Status on Column Headers** if you want the the sort button to be always displayed on the table column headers with the current sort status. If this option is unchecked, each time you want to get the sort button for a column, you need to put the mouse pointer on its header to make the button shown.
    4. Run a dashboard containing a table and you can click on the table column headers to sort data.

       To sort the table based on any column values ascendingly, click the upward triangle on the column; to sort the values descendingly, click the downward triangle. When a sort order is applied, the corresponding triangle will be highlighted. To remove the sort order, click the highlighted triangle.
- **Filtering data in a table**
  - **Using shortcut menu**
    You can use filter-related commands on the shortcut menu to filter data in a table. To do this, right-click on any value of a detail field by which you want to filter data, then you can see the Filter item which provides a submenu containing the following commands:
    - **Remove Filter**
      This command is enabled after you have applied filtering on the field. Clicking this item will remove all filters on this field.
    - **First N**
      Shows the First N item with which you can filter data to display records that meet the First N condition. You can select a number from the submenu or enter a positive integer into the text box on the submenu to specify the First N condition.

      For example, if you select 5 from the First N submenu for a certain field, then only the records with the field value equal to one of the first five field values will be displayed.
    - **Last N**
      Shows the Last N item with which you can filter data to display records that meet the Last N condition. You can select a number from the submenu or enter a positive integer into the text box on the submenu to specify the Last N condition.

      For example, if you select 5 from the Last N submenu for a certain field, then only the records with the field value equal to one of the last five field values will be displayed.
    - **Field values**
      "Field values" is not the name for a command on the Filter submenu, but represents some items which are the values of the field you right-click on. Selecting any field value listed here will make the table only display records with the field value equal to the selected one.
    - **More**
      This command is enabled if the Filter submenu cannot list all field values. When it is enabled, clicking it will show the Select Values dialog, with which you can search for and select values conveniently with the help of the quick search toolbar.
  - **Using labels**
    You can use a label to control the filter conditions in tables. This feature needs to be enabled at report design time.
    1. In JReport Designer, select a label in a table in a library, and then set its Filterable property to true.
    2. Set the field by which you want to filter records as the value of the label's Bind Column property.
    3. Save the library component and publish it to JReport Server.
    4. Create a dashboard and insert the table in it, and you can find that a filter button displayed beside the label. Click it to show the Filter list, which contains the same items as the Filter submenu. Click the corresponding item to filter the records.

       After a filter is applied on the field specified by the Bind Column property, the button will be highlighted. You can still click it to show the Filter list to change the filter condition or select the All item to remove the filter on the field.
  - **Using server profile setting**
    You can also configure the server profile to enable filter records in tables by clicking the filter button on table column headers.
    1. Open the Profile > Customize Profile > Common tab.
    2. Check the option **Filter on Column Headers** to enable clicking the filter button on column headers to filter table records.
    3. Check **Show Sort/Filter Status on Column Headers** if you want the the filter button to be always displayed on the table column headers with the current filter status. If this option is unchecked, each time you want to get the filter button for a column, you need to put the mouse pointer on its header to make the button shown.
    4. Run a dashboard containing a table and you can click on the table column headers to filter data.

       To filter the table based on any column field, click on its column header to show the Filter list, which contains the same items as the Filter submenu. Click the corresponding item to filter the records.

       After a filter is applied on a field, the button on its column header will be highlighted. You can still click it to show the Filter list to change the filter condition or select the All item to remove the filter on the field.

Notes:

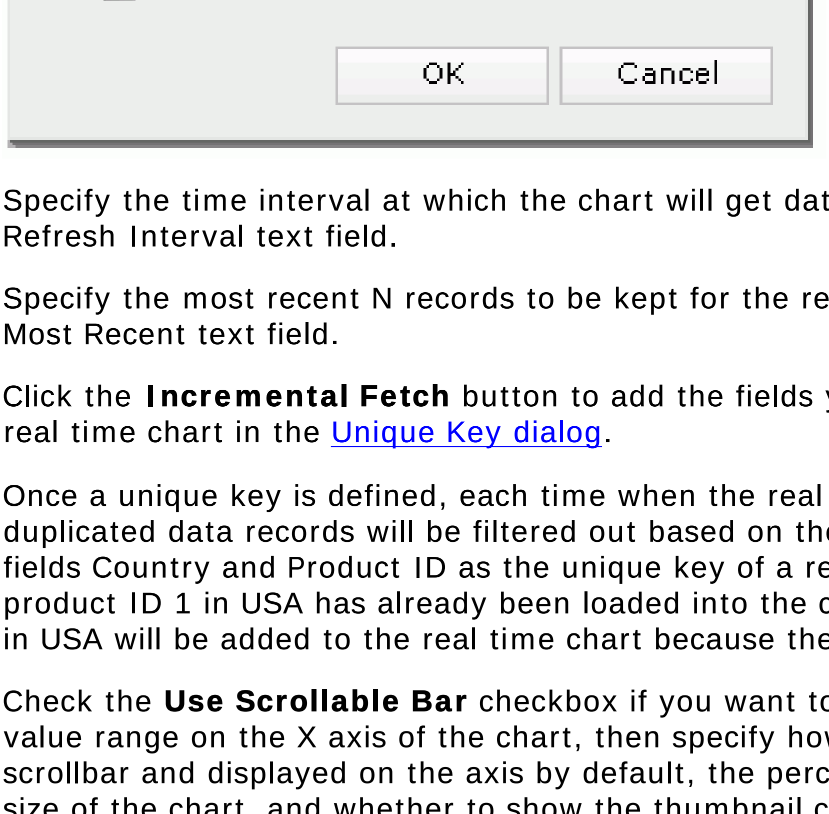
- When sorting the data in a table by a field and then by another field, the later sort condition is the major order, and the former condition the minor order, that is to say, the records in the table will be sorted by the later sort condition first, then by the former sort condition.
- The Sort on Column Headers and Filter on Column Headers features do not work on summary columns that contain multiple summary fields.
- When a table is designed with the Sort by Label/Filter by Label feature in JReport Designer, if the Sort on Column Headers/Filter on Column Headers option is also enabled on the server profile, the settings defined in JReport Designer have the higher priority for the specified columns.

Manipulating a crosstab

- **Sorting column/row header**
  To sort a column/row header, right-click on the header, then on the shortcut menu, select **Ascend** or **Descend** from the Sort submenu. To remove the sort condition, select **No Sort**.
- **Switching crosstab fields**
  A convenient way to change the fields in a crosstab is by using the Switch command. To do this, right-click a row/column/summary and select **Switch Row**/**Switch Column**/**Switch Summary** from the shortcut menu. The fields available for the row/column/summary are listed in the submenu with the current used field checked. Select the field you want to use to replace the current one.
- **Adjusting the width of crosstab fields according to the contents**
  When the contents in the field of a crosstab need more space to completely display, you can adjust the width of the field according to its contents. To achieve it, right-click the field and select **Autofit** from the shortcut menu.

Manipulating a chart

- **Swapping chart groups**
  You can switch data between the category and series axes, or between the category and value axes of a chart if no field on the series axes. To do this, right-click in the chart, then on the shortcut menu, click **Swap Chart Groups**.
- **Sorting category/series labels**
  You can sort the labels on the category or series axes of a chart in either descending or ascending alphabetical order. To do this, right-click in the chart, then on the shortcut menu, select the required order from the Sort Category or Sort Series submenu.
- **Changing the chart type**
  Right-click in the chart, then on the shortcut menu, locate **Chart Type**. From the drop-down menu, select the desired chart type and its subtype.
- **Zooming in chart values**
  For a bar, bench, line, area or stock chart, you can select the values you are interested in to have them zoomed in. To do this, drag the mouse from the start value to the end value you want to select, then release the mouse. The selected values will be zoomed in. If you want to return to the initial status, just click the return button on the upper right corner of the chart paper.
- **Showing/Hiding the chart legend or legend label**
  Right-click the chart and select the corresponding show/hide option from the shortcut menu to show or hide the legend or legend label. However, the legend will always be displayed at the export and print result.
- **Configuring real time chart settings**
  For a real time chart, you can further configure it when inserted in a dashboard. To do this:
  1. Right-click the chart and select **Real-time Settings** from the shortcut menu. The Real-time Settings dialog appears.

  2. Specify the time interval at which the chart will get data and refresh itself automatically in the Refresh Interval text field.
  3. Specify the most recent N records to be kept for the real time data on the chart in the Show Most Recent text field.
  4. Click the **Incremental Fetch** button to add the fields you want to use as the unique key of the real time chart in the Unique Key dialog.

     Once a unique key is defined, each time when the real time chart automatically updates itself, duplicated data records will be filtered out based on the unique key. For instance, if you add the fields Country and Product ID as the unique key of a real time chart, when a record with the product ID 1 in USA has already been loaded into the chart, no more records of this product ID in USA will be added to the real time chart because they have the same unique key value.
  5. Check the **Use Scrollable Bar** checkbox if you want to use a scrollbar to control the visible value range on the X axis of the chart, then specify how many data items will be selected on the scrollbar and displayed on the axis by default, the percentage the scrollbar occupies the whole size of the chart, and whether to show the thumbnail chart in the scrollbar.
  6. Click **OK** to accept the settings and close the dialog.
- **Stopping or resuming a real time chart from refreshing**
  For a real time chart, you can stop or resume it from automatically refreshing by right-clicking it and selecting the Pause Refresh or Resume command from the shortcut menu.

Manipulating geographic map group markers/areas

- **Going up/down on geographic map group markers/areas**
  - For the group level that is higher than some other group levels in a geographic map component, point to its group marker/area, right-click it and select **Go Down** from the shortcut menu to jump one group level down.
  - For the group level that is lower than some other group levels in a geographic map component, point to its group marker/area, right-click it and select **Go Up** from the shortcut menu to jump one group level up.
- **Showing/Hiding geographic map group markers/areas**
  By default, all visible group markers/areas are shown. To hide them, right-click the geographic map (not the group markers/areas) and select **Hide Markers/Areas** from the shortcut menu. If you want to show them again, right-click the geographic map and select **Show Markers/Areas** from the shortcut menu.

Filtering component data

When creating or editing dashboards, you can filter component data using two types of web controls: sliders or filter controls.

Filter controls allow you to pick one or more random values from a list and are used with categorical or nominal variables. You can choose one or more values from anywhere in the list and there is no mean or median value calculation possible such as States and Countries.

Sliders allow you to pick one or more sequential values from a list and are used for interval variables such as dates, times, quantity and currency variables where the slider represents the scale from lowest to highest value and the middle represents the median value.

Think about what you need and choose the proper filtering tool.

Filtering scenarios

Both sliders and filter controls have the same filtering mechanism.

Filtering based on one field is a common usage. Bind a field to a slider or filter control, and then based on the field to filter the data of the components created from the same data source as the field.

Another special usage is to filter components using different data sources. Choose a common field all the data sources contain and then bind a slider or filter control with the common field in all the data sources, that is, in the insertion dialog, select the common field under the nodes of all the data sources.

For example, there are two components containing data from different data sources, and you want to filter their data using one filter tool. The precondition is that both components have the field you want to filter. For example, you would like the two components to show the data of a specific country. In order to do this, insert a slider or a filter control according to your requirement into the dashboard body, then in the insertion dialog, select both the country fields from the two data sources (different data sources may use different names for the country field, for example, data source 1 uses "Country", data source 2 uses "P_Country", in this case, you need to select both "Country" and "P_Country"), then from the Apply To drop-down list, select the two components you want to filter. The filter tool will be inserted in the dashboard body, and you can see it lists country names which come from the two data sources. In the filter tool select one or more countries, then the two target components will be filtered and only display the data of these selected countries.

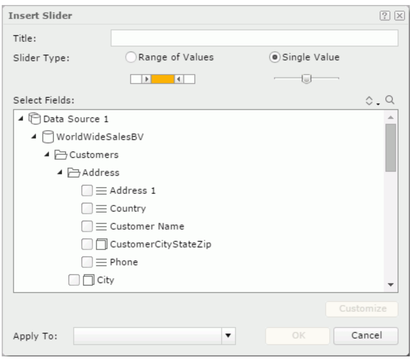
When you bind a slider or filter control with multiple different fields, be sure the list of values in each field match so that when you select a value, such as Country, P_Country and S_Country, it will match the appropriate country field in each component. The logic is the values are OR that is *Field1=SelectedValue1* or *Field2=SelectedValue1* or *Field3=SelectedValue1*. Therefore, when Field1, Field2 and Field3 are used in different components you will see the records correctly in each component.

When you bind a slider or filter control with multiple different fields which do not contain the same list of values such as Country, Region and Territory, at runtime after you select values in the slider or filter control, the filter condition will use OR logic to apply the selected values to all the fields of the slider or filter control, for example, *Field1=SelectedValue1* or *Field2=SelectedValue1* or *Field3=SelectedValue1*. In this case, when there are three fields but the list of values in each field do not match, the result will have no matching records for two of the components and therefore become blank components. We recommend that you use three different filter tools in cases like this.

Using sliders

To insert a slider to the dashboard body:

1. Click **Show Resources** on the side bar to display the Resources panel.
2. From the Toolbox node, drag **Slider** to the destination in the dashboard body. The Insert Slider dialog appears.



3. In the Title text field, input a title for the slider.
4. Choose whether the slider is used to specify a range of values or just a single value.
5. From the resource list, select the fields of the same data type to bind to the slider.

   To filter components created from the same data source, select a field in the data source.

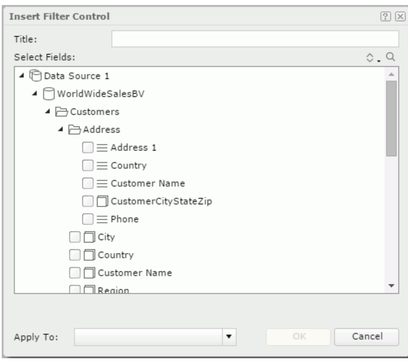
   To filter components created from different data sources via one slider, find a common field these data sources contain, then select the field in each of the data sources.
6. By default, all the values of the selected fields will be available on the slider, which may be too many for a slider. However you can customize the values you want to show on the slider. To do this,
   a. Click the **Customize** button to display the Customize Value dialog.
   b. Uncheck the **All Values** option. You can then customize the values.
      - To specify values one by one, click + to add a value line, then in the line to type a value or select a value from the drop-down list. Repeat the operation to add more values. To remove a value, select its line and then click the × button.
      - To specify the value range, specify a value in the From and To text boxes respectively.
   c. For fields of the Date/Time type, you can also specify a special function.
   d. Click **OK** to save the customized values. The customized values will be available on the slider for choosing.
7. The Apply To drop-down list provides the components involving the selected fields. Select the components which you want to filter.
8. Click **OK**. A slider bound with the specified fields will be inserted in the dashboard body. You can then specify values in the slider to filter the specified components.

You can further modify the sliders inserted into a dashboard if you want. To do this, hover the mouse over the title bar of a slider, then click ▾ that appears on the title bar and select **Edit Setting** from the drop-down list. In the Edit Slider dialog, edit the setting as required.

Using filter controls

To insert a filter control to the dashboard body:

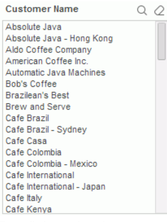
1. Click **Show Resources** on the side bar to display the Resources panel.
2. From the Toolbox node, drag **Filter Control** to the destination in the dashboard body. The Insert Filter Control dialog appears.



3. In the Title text field, input a title for the filter control.
4. From the resource list, select the fields of the same data type to bind to the filter control.

   To filter components created from the same data source, select a field in the data source.

   To filter components created from different data sources via one filter control, find a common field these data sources contain, then select the field in each of the data sources.
5. The Apply To drop-down list provides the components involving the selected fields. Select the components which you want to filter.
6. Click **OK**. A filter control bound with the specified fields will be inserted in the dashboard body. You can then specify values in the filter control to filter the specified components.



   To easily locate the values you want in the filter control, you can make use of its quick search toolbar. To display the toolbar, click on the field name bar. The button on the field name bar is used to cancel the selection of values in the filter control. For detailed usage about the quick search toolbar, refer to Select Values dialog.

You can further modify the filter controls inserted into a dashboard if you want. To do this, hover the mouse over the title bar of a filter control, then click ▾ that appears on the title bar and select **Edit Setting** from the drop-down list. In the Edit Filter Control dialog, edit the setting as required.

Notes:

- When there are more than 300 values in a filter control, JReport will use Big Data Loading logic. In this case, the Shift Key for multiple selection will not work.
- Sometimes, when the position of a filter control is changed due to layout change, the advanced search toolbar in the filter control may not follow the filter control but stay where it is.

Specifying parameter values

To specify parameter values to a dashboard:

Click on the side bar to display the Enter Parameter Values dialog which lists all the parameters used in the current dashboard. The way to specify a parameter value varies with the type and properties of the parameter. Here are several ways you can use to specify parameter values:

- In the parameter value combo box, input the value manually or select the required one from the drop-down list. If you have chosen to automatically save parameter values, the saved value groups will be available on the top of the parameters' value lists for selection.
- Select or unselect the checkbox to specify a Yes/No value.
- Click the button to specify multiple values in the Enter Values dialog.
- Click the calendar button to specify a date and time value using either calendar or expression in the Calendar dialog. If you use an expression to specify the value, when you hover the mouse pointer over this value, a tip will appear showing its expression. Click on the value, the expression will be displayed in the value text box and you can edit the expression in the text box directly if you want. After editing, when you click elsewhere outside of the value text box:
  - If the edited expression is correct, a new value calculated by the expression will be displayed in the parameter value text box.
  - If the edited expression is wrong, no value will be created and the expression itself will be displayed and highlighted in red in the value text box. Correct the expression, or click the calendar button to specify a value.
- Click the **Use Saved Values** button and select a previously saved parameter value group to apply to the dashboard. For details, see Manually saving parameter values.

To specify parameter values to a library component:

You can access the configuration panel of a library component to specify its parameter values. To do this, hover the mouse over the title bar of the library component, then click that appears on the title bar and select **Edit Setting** from the drop-down list to display the configuration panel. Specify the values in the panel and click **OK** to apply them to the library component. The Cancel button is used to close the configuration panel.

The Show Parameters Before Running option in the panel is used to control whether to show the configuration panel **before** rendering the library component each time the dashboard is run or refreshed. For example, if a library component uses parameters, and you would like to specify parameter values before loading the component rather than first loading the component and then changing the parameter values so as to run the component the second time to get the desired result, you can check this option to have the panel displayed before rendering by default.

Note: You are recommended not to use blank as the thousands separator in Number-typed parameter values under French locale, otherwise your input will not be correctly recognized because of a JVM bug. For details, see http://bugs.sun.com/view_bug.do;jsessionid=c8cdaf911b20ffffffffffd9fc6340b30d670?bug_id=4510618.

This document also introduces some useful techniques for setting parameter values.

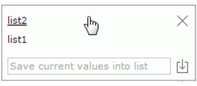
Saving parameter values for reuse

When specifying parameter values for dashboards, you may want to save the specified parameter values for reuse next time. JReport provides two ways of saving. One is users decide when and which parameter values to save, the other is JReport saves each applied or submitted parameter values automatically. When the saved number in either way reaches the maximum, the oldest record will be removed. The number is calculated on a user-dashboard basis. Take a dashboard with two parameters for an example, supposing the maximum number is set to 3. Each user can save at most three groups of parameter values for the dashboard. Each group contains the values of the two parameters.

To switch on the function of saving parameter values, go to the Profile > Customize Server Preferences > Advanced tab, select **Yes** to the option Enable Saving Parameter Values, select a way to do the saving: Manually or Automatically, and then specify a maximum number to limit the saved value groups which is called the auto complete parameters list for each user-dashboard pair.

Manually saving parameter values

When you have chosen to manually save parameter values, the Use Saved Values button will be available in the Enter Parameter Values dialog. After clicking the button, a drop-down list that contains the lists of previously saved parameter values will be displayed for you to choose one to apply.



The button next to a drop-down list is used to delete the saved list from the list library.

You can also save the current parameter values as a whole marked as a list for reuse next time, by typing a name for the list in the text box which shows "Save current values into list" and then clicking . The parameter value lists saved for a dashboard are limited. The maximum number is controlled by the option Maximum Number of Auto Complete Parameters List in the Profile > Customize Server Preferences > Advanced tab. By default it is 3. When the number of the saved parameter value lists reaches the maximum number, if you want to save another parameter value list, it will overwrite the oldest one.

Using automatically saved parameter values

When you have chosen to automatically save parameter values, each time a user submits a group of parameter values to a dashboard, the group is saved by JReport automatically. The next time the same user runs the dashboard, the auto saved parameter values will be available in the parameters' value lists for selection in the Enter Parameter Values dialog.

Customizing default parameter values

When specifying parameter values for a dashboard, if you would like the current specified parameter values to be the default selected values in the Enter Parameter Values dialog next time when you run the dashboard, select the **Save as default** option in the dialog (to make this option available, you need to make sure the Enable Setting Default Parameter Values For Dashboard is selected in the Profile > Customize Server Preferences > Advanced tab).

The save as default option is a user-dashboard level setting, that is to say, it takes effect when both the same user and dashboard are matched. This also applies to admin users, and therefore admin cannot customize the setting for all users.

Sharing parameters between library components

A dashboard can have multiple library components and all these components are independent from each other. When several components contain parameters which have the same name and come from the same catalog and at the same time contain exactly the same values, by default these parameters are regarded as separate parameters each connected to its own component. To submit a value to all these components, you need to submit the value to each of the parameters. However by sharing the parameters, you only need to provide a value for one of them as a representative of all the parameters. You need then only submit values to this parameter and all the components will use the specified values. By sharing parameters, the same parameters will only send query once to the database, thus the dashboard performance can be improved.

What parameters can be shared

JDashboard supports sharing two types of parameters coming from different library components in the same dashboard:

- **The same parameters**
  The parameters with the same name and coming from the same catalog.

  For example, a parameter *@Category* comes from the catalog A.cat, and in a dashboard it is used by two library components LC1 and LC2. So in the dashboard they are two parameters marked as *LC1.@Category* and *LC2.@Category*. These two parameters are the same parameter and only require the user to specify one value for them.
- **The same meaning parameters**
  The parameters that are not the same parameters but have the same parameter type and value data type. For cascading parameters they should have the same hierarchies and each hierarchy have the same parameter value type, then they can be shared as parameters with the same meaning even though the names are different.

  For example, there are two parameters *@Province* and *@State* which have the same parameter type and value type so they can both have the same meaning. For the cascading parameters like *(@Country, @Province)* and *(@Country, @State)* which have the same hierarchies and each hierarchy has the same parameter value type so they also can have the same meaning.

The rule for merging the values of the shared parameters

Within a sharing group, if all the shared parameters allows for type-in values, the merged value result is the union of the values of all the parameters. The merged value names are distinct. However if any of the shared parameters does not allow type-in values, the merged value result is the intersection of all the values.

Unexpected results after sharing parameters

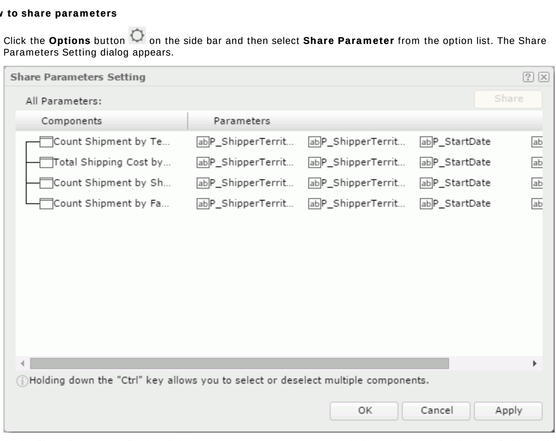
Sometimes there may be unexpected parameter sharing that does no harm to the report system and report data, but the dashboard result may be unexpected.

After sharing parameters, their values will be merged. The merged values result may be bigger or smaller than the value lists of some of the shared parameters, which might lead to some values cannot be supported by some components or some values can never be available to some components.

For example in the case of @Province and @State, if the parameters are shared then the list of values will be all states and provinces. However, one library component may use a query that limits the data to US only, thus if the user selects a Canadian province from the list the component will have no data.

How to share parameters

1. Click the **Options** button on the side bar and then select **Share Parameter** from the option list. The Share Parameters Setting dialog appears.



2. In the Share Parameters Setting dialog, select the parameters you would like to share by holding the Ctrl key. If the selected parameters support being shared, the Share button will be enabled. Click the button. The parameters will be added into one sharing group.

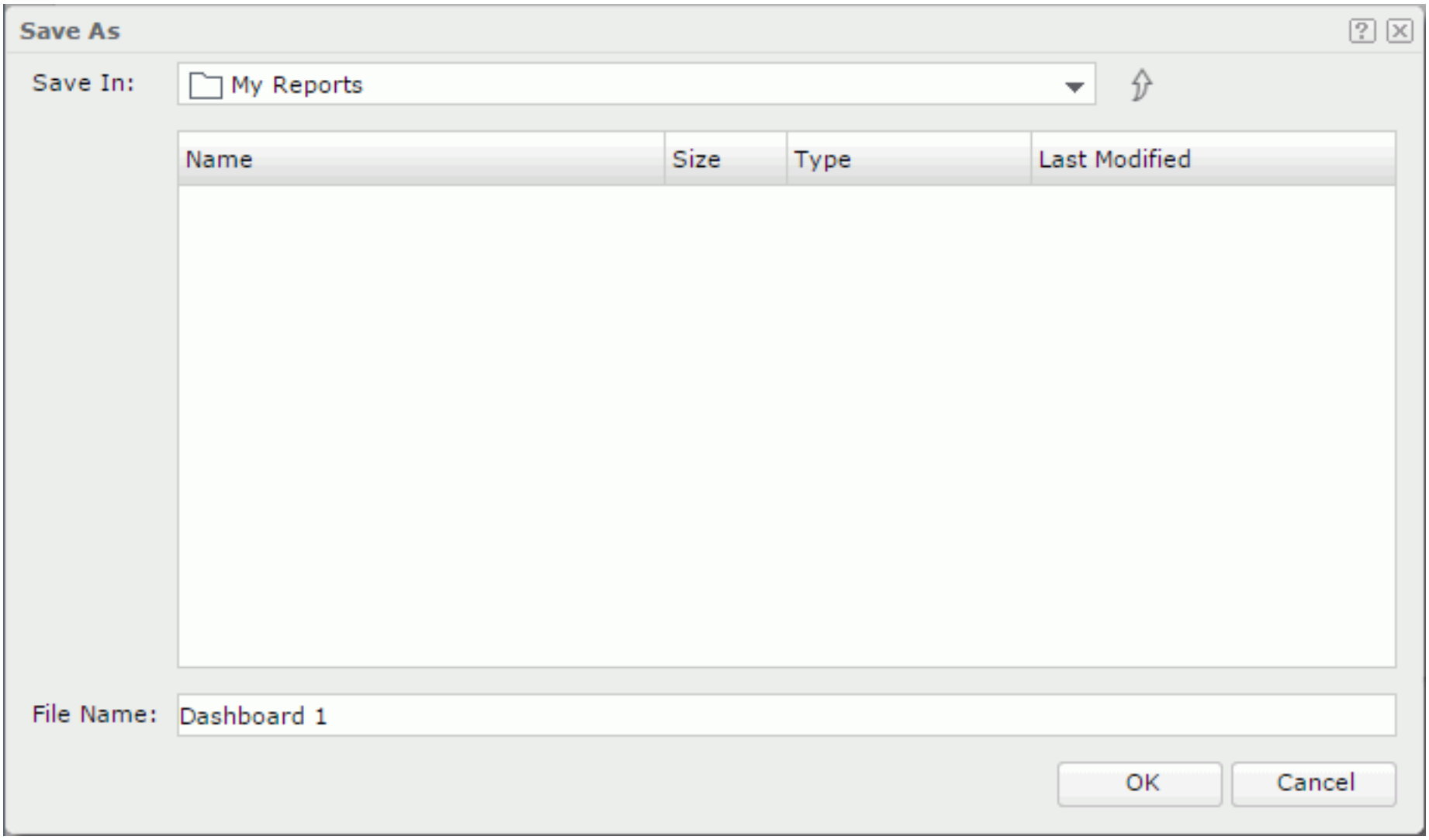
   To add another parameter into a sharing group, select any parameter in the group while holding Ctrl and then select the parameter, then click **Share**.

   To remove parameters from a sharing group, select the parameters and then click **Cancel Share** which appears in the place of the Share button.
3. Click **OK** to finish.

Saving dashboards

To save the changes you made to the current dashboard, click the **Save** button on the side bar, or click the **Options** button and select **Save** from the option list.

If the dashboard is newly created and has not yet been saved, the Save As dialog appears.



1. In the Save In section, browse to the folder where you want to save the dashboard in the server resource tree. The folder may be Public Reports or My Reports. You can use the button to return to the parent folder.

   The resource table shows the resources in the current directory. Click the column names to change the order of the report in the table list if required.

2. In the File Name box, enter the name of the dashboard or use the default name.
3. Click **OK** to save the dashboard.

To save a copy of a dashboard, click on the side bar and select **Save As** from the option list to show the Save As dialog, and then do as above.

After saving your dashboard into the server resource tree, you can browse to its directory on the JReport Console > Resources page and run it directly just like you run a report.

To find a newly saved dashboard version, browse to select the row that the dashboard is in on the JReport Console > Resources page, click **Tools** > **Version** on the task bar, the Dashboard Version panel will be displayed showing the versions.

Note: You will not be able to save the dashboard to some locations if you do not have Write permission.

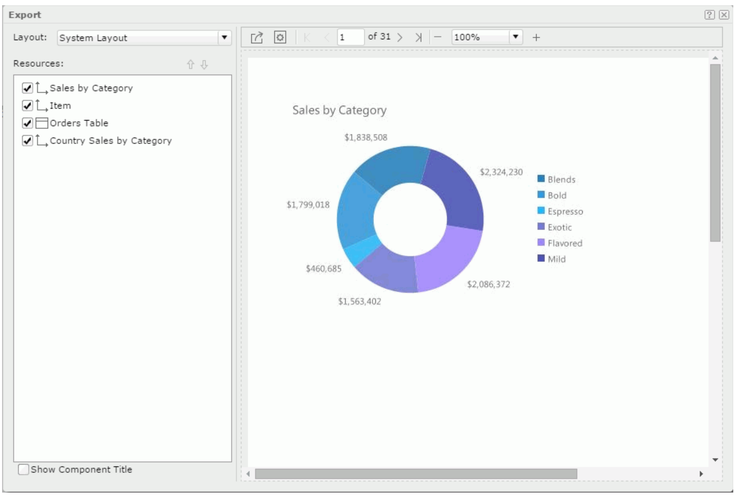
Exporting library components

Library components visible in the page panel can be exported, for example, library components that were created using JReport Designer and inserted into dashboards via the Resources panel.

To export the library components in a dashboard, click the **Export** button on the side bar, or click the **Options** button on the side bar and select **Export** from the option list. Then in the Export dialog, choose in which layout you want the library components to be exported, system layout or custom layout. In the system layout, JReport will calculate the positions of the library components in the dashboard following certain rule. If you are not satisfied with the system layout, you can customize the layout by yourself to determine the position of each library component and specify whether to export all the data of a table/crosstab or just the current page of the table/crosstab displayed in the dashboard. The custom layout can be saved for all users who can access the dashboard.

To export using system layout:

1. In the Export dialog, select **System Layout** from the Layout drop-down list.

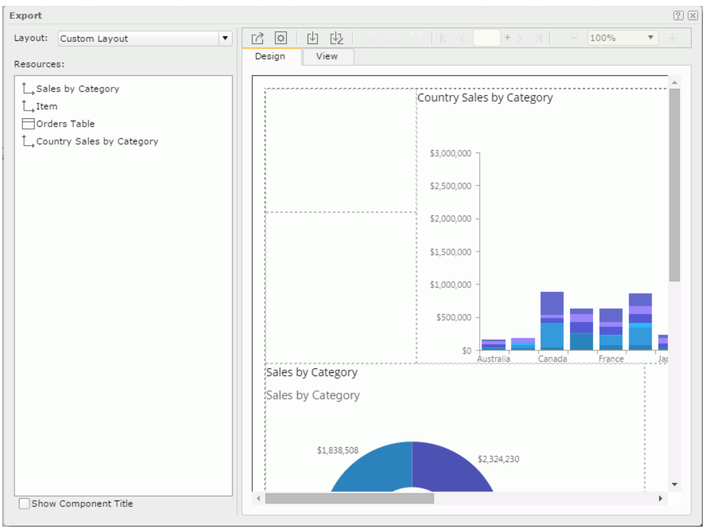


   The preview of the layout is displayed in the preview panel on the right. Sometimes you may find that the layout is a little different from what you see in the dashboard.

2. In the Resources box, select the library components you want to export. By default all exportable library components are listed and selected.
3. The order of the library components in the Resources box determines the order in which they will be exported. Click and to adjust the exporting order if necessary.
4. Check the **Show Component Title** option if you want to show the library component titles in the exported result.
5. Click the page navigation buttons on the toolbar of the preview panel to browse the pages if you want.
6. Click the **Page Setup** button on the toolbar to set the page properties of the exported result file such as page size and margins in the Page Setup dialog.
7. Click to do the last setting and then export.
   a. From the Export File Format drop-down list, specify the format for exporting the library components. For a format other than XML, all the selected library components will be exported into one single file. For XML, each library component will be exported to a separate XML file. When the exported result contains more than one file, all the files will be zipped.
   b. In the File Name text field, specify a name to the exported result, which could be either a single file or a zipped package name.
   c. Check **Run Linked Report** if you need to include linked report in the exported result file.
   d. Click **OK** to start exporting.

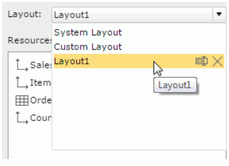
To export using custom layout:

1. In the Export dialog, select **Custom Layout** from the Layout drop-down list. You can also select an existing custom layout to modify it.



2. Check the **Show Component Title** option if you want to show the library component titles in the exported result.
3. Click the **Page Setup** button on the toolbar to set the page properties of the exported result file such as page size and margins in the Page Setup dialog.
4. In the Design tab, customize the layout of the library components by making use of the following operations. The library components are arranged using a tabular with each cell holding one component.
   - Split or merge cells using the toolbar options: , , and .
   - Drag a library component from the Resources box into a blank cell in the Design tab. A cell can hold only one library component.
   - Delete a library component from a tabular cell, by using the shortcut menu option Remove.
   - For a table or crosstab, only its current view of data as displayed in the dashboard will be exported by default. If you want its full data to be exported, right-click on the table or crosstab and select **Filter** from the drop-down list. Then in the Filter dialog, switch from Current View to **All** and click **OK**.
5. Click the **View** tab to preview the layout. You can browse the pages and zoom in/out by clicking the corresponding toolbar buttons.

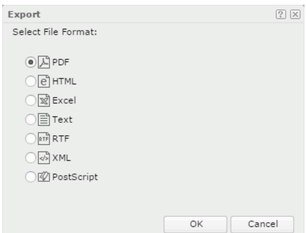
   If you are satisfied with the layout and want to save it for future use, click or on the toolbar, then in the Save As dialog, provide a name for the layout and click **OK**. The saved custom layouts will be available in the Layout drop-down list to all users who can access the dashboard. For each of them, you can edit its name or delete it using the two buttons - Rename and Delete - appearing on the right when the mouse hovers over the layout item on the drop-down list.



6. Click , then in the Export dialog, do the last setting and start exporting.

Tip: If you just want to export a single library component, there is a more convenient way.

1. Hover the mouse over the title bar of the library component, then click that appears on the title bar and select **Export** from the drop-down list. The Export dialog appears.



2. In the dialog, choose the format to which you want to export the library component, then click **OK**.

However, using this way, if the library component is linked with other reports, you cannot control whether to generate the linked reports while exporting. The linked reports will not be included in the exported result for all time.

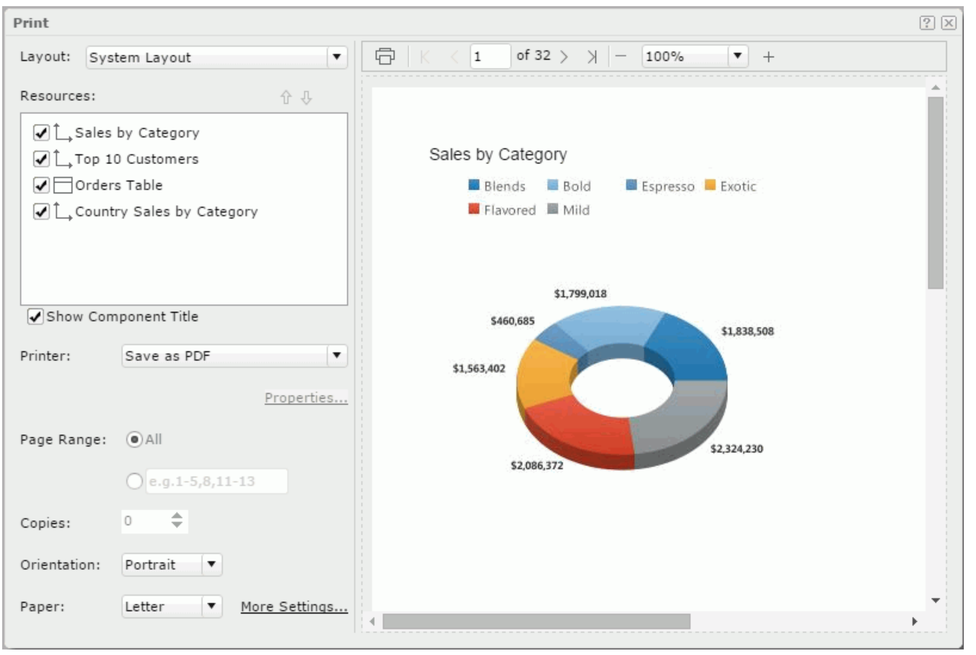
Printing library components

Library components visible in the page panel can be printed, for example, library components that were created using JReport Designer and inserted into dashboards via the Resources panel.

To print the library components in a dashboard, click the **Print** button on the side bar, or click the **Options** button on the side bar and select **Print** from the option list. Then in the Print dialog, choose in which layout you want the library components to be printed, system layout or custom layout. In the system layout, JReport will calculate the positions of the library components in the dashboard following certain rule. If you are not satisfied with the system layout, you can customize the layout by yourself to determine the position of each library component and specify whether to print all the data of a table/crosstab or just the current page of the table/crosstab displayed in the dashboard. The custom layout can be saved for all users who can access the dashboard.

To print using system layout:

1. In the Print dialog, select **System Layout** from the Layout drop-down list.

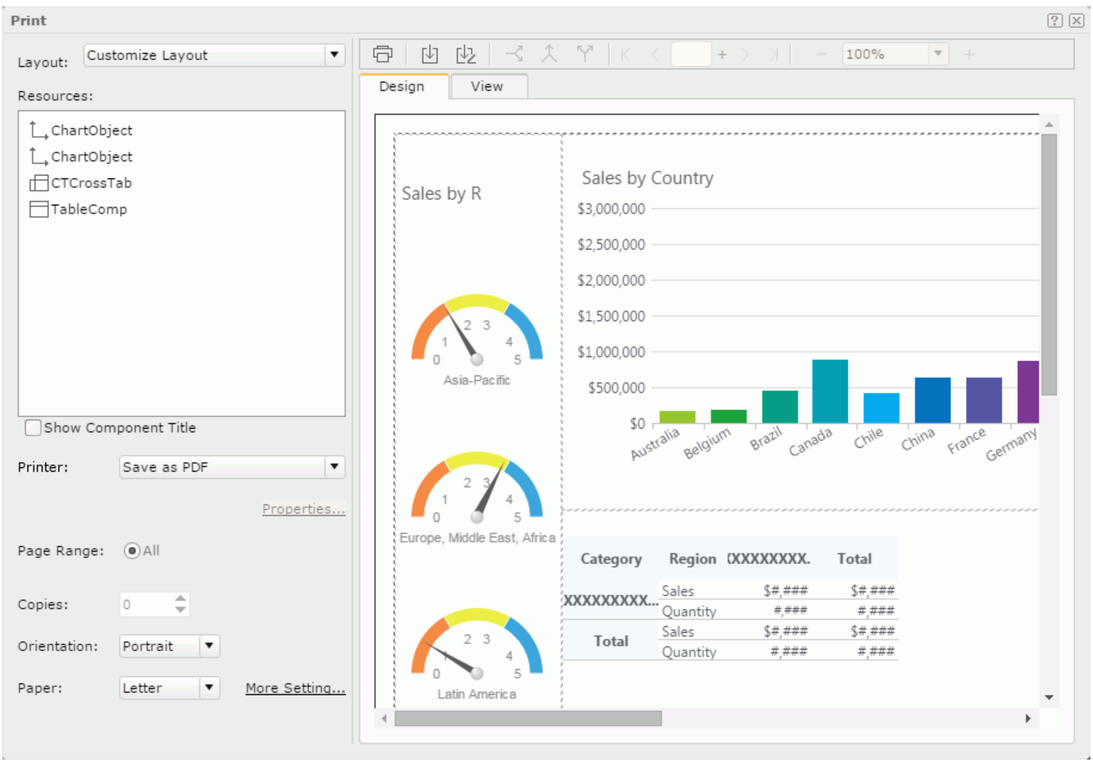


   The preview of the layout appears in the preview panel on the right. Sometimes you may find that the layout is a little different from what you see in the dashboard.

2. In the Resources box, select the library components you want to print. By default all printable library components are listed and selected.
3. The order of the library components in the Resources box determines the order in which they will be printed. Click and to adjust the printing order if necessary.
4. Check the **Show Component Title** option if you want to show the library component titles in the printed result.
5. From the Printer drop-down list, select the printer using which to print the library component result. All the printers the web browser can access will be listed here. If you want to download the result to a PDF or HTML file and then use your local printer to print the file, select **Save as PDF** or **Save as HTML**.
6. If a printer is selected, click the **Properties** link to specify the printing properties in the Printer Properties dialog, then specify the number of copies you want to print accordingly.
7. Specify the page orientation and paper type for the printed result. If necessary, click the **More Settings** link to set more page properties such as paper size and margins in the Page Setup dialog.
8. Click the page navigation buttons on the toolbar of the preview panel to browse the pages if you want.
9. Click on the toolbar to start printing. If Save as PDF or Save as HTML is selected in the Printer drop-down list, the result file of the library components is then opened in an associated program with which you can print the result to a printer.

To print using custom layout:

1. In the Print dialog, select **Custom Layout** from the Layout drop-down list. You can also select an existing custom layout to modify it.



2. Do the same as described in steps 4 to 8 of the above system layout procedure.
3. In the Design tab, customize the layout of the library components, then preview it in the View tab. For how to customize the layout, refer to Exporting using custom layout.
4. Click on the toolbar to start printing. If Save as PDF or Save as HTML is selected in the Printer drop-down list, the result file of the library components is then opened in an associated program with which you can print the result to a printer.

Setting JDashboard as the server home page

Setting JDashboard as the JReport Server user console home page allows for a faster and easier access to frequently visited dashboards with their latest data. The feature saves JDashboard status as the contents of the home page, such as which dashboards are open and which dashboard is active. Then you will be able to easily access the JDashboard with the saved status by one of the following ways:

- Log onto the JReport Console page and you will be directed to the Home tab right away which displays the JDashboard. The Home tab is available after you have set JDashboard as the server home page.
- Click the Home tab on the system toolbar of the JReport Console page to switch to the JDashboard.

To set JDashboard as the server home page:

1. In the Profile > Customize Server Preferences > General tab on the JReport Administration or JReport Console page, set the option Use JDashboard as Server User Console Home Page to **Yes** to enable setting JDashboard as server home page.
2. In JDashboard, open the dashboards that you are going to view a lot and make sure they have been saved. Focus on the dashboard you would like to see first once the JDashboard is loaded, then click the **Options** button on the side bar and you will see Set as Server Home is enabled on the option list. Click the option to set the current JDashboard status as the home page.

In the same JDashboard window, you can set the home page at any time you want. The last time you do this before exiting JDashboard will take effect.

To cancel setting the home page, in the Profile > Customize Server Preferences > General tab on the JReport Administration or JReport Console page, uncheck **Yes** for the option Use JDashboard as Server User Console Home Page.

Running dashboards via URL

Similar to reports, dashboards can be created or run directly from your application. JDashboard uses /{context_root}/dashboard/app/entry/run.jsp as the URL entry. Here {context_root} is the servlet's context root when deployed in a WAR or EAR file as a servlet.

Creating a new dashboard

To create a new dashboard from URL, simply use /{context_root}/dashboard/app/entry/run.jsp.

For example, run the URL http://localhost:8888/dashboard/app/entry/run.jsp, and JDashboard will be opened with a new blank dashboard in it. You can then begin to build your dashboards.

Visiting JDashboard as the server home page

After JDashboard has been set as the server user console home page, you can use a URL to access the home page.

Parameter name: jid_lastsession
Parameter value: true/false
Examples:

- http://localhost:8888/dashboard/app/entry/run.jsp?jid_lastsession=true
- http://<IP>:<port>/jreport/dashboard/app/entry/run.jsp?jid_lastsession=true, here jreport is the servlet context root.

Opening specific dashboards

JReport provides parameters for developer users to run dashboards via URLs. Some parameters are encapsulated as JSON (JavaScript Object Notation) objects. Therefore, it will help if you obtain some knowledge of JSON before using the syntax more clearly. When composing the URL, you need to use URL encoding to avoid errors. To encode URLs by JavaScript, the function encodeURI is recommended.

- jid_dashboard_mode
"active" :0, // The index number of the dashboard that will be active once the JDashboard is loaded. 0 means the first dashboard specified in the value list of the parameter "resist", 1 the second dashboard, 2 the third, and so on.
"resist":[
// Specify the dashboards to open. The order of the dashboards specified here reflects the order they are displayed in JDashboard.
{
{"name":"DashboardFileNameWithFullPath", // The dashboard file name with its full path. The path could be a disk path or a server resource tree path. For the latter, it could be either in the My Reports folder (/USERFOLDERPATH/admin/) or in the Public Reports folder (/).
"ver":"-1", // Optional: The version number of the dashboard/report. -1 means the latest version.
"real":"false", // Optional. Value: true/false. true means the resource path is a real disk path and false means a server resource tree path. When it is a real path, "real":"true" must be specified.
"type":"40", // Indicates it is a dashboard.
"param_pap":"true", // Optional. Value: true/false. Specify whether to pop up the parameter dialog if the dashboard uses parameters.
"fch_parameter":{
// Specify the parameter values used in the dashboard.
{
{"lc_names":["lc1","lc2",...}, // Optional: lc1 and lc2 can be either of the following:
- The names of the library components in the dashboard, with full path in the server resource tree, either in the My Components folder (/USERFOLDERPATH/admin/) or in the Public Components folder (/COMPONENT_LIB/). For example, /COMPONENT_LIB/SampleReports/Sales Bar.lc.
- The IDs of the library components. The ID of a library component can be known in the library component tool panel. To access the panel, hover the mouse over the components in the bar, then click "". that appears on the title bar and select About from the drop-down list.

"jrd_params":{"parameter1":"value1,"parameter2":
{"value1","value2",...} // If lc_names and jrd_params are specified together, the parameters here are applied to the library components given by lc_names, otherwise to the dashboard. When both dashboard level and library component level parameters exist in the URL, the order to apply parameters will be: the dashboard first, and then the library components. When several library components are the same in the dashboard, the parameters specified to one of them will also work on the others. In the case when the URL does not specify parameters for a specific library component but the dashboard level parameters are available, if later a new library component is added into the dashboard, it will use the parameter values in the URL. When a library component is set to use a new connection in the URL, then when adding it into the current dashboard, the library component uses the new connection.
}
}
},
... // Specify more dashboards following the above format.
]
}
}
- jrd_dashboard_mode
// Specify whether JDashboard runs in the edit mode or view mode. The value is "edit" or "view". When this parameter is not provided in the URL, the default mode uses the one set in the server preference.

Examples:

- http://<IP>:<port>/jreport/dashboard/app/entry/run.jsp?jrd_resext=
{"active":0,"resist":[{"name":"/USERFOLDERPATH/admin/Dashboard 1.dsh","ver":"-1"}]}, here jreport is the servlet context root.
- http://localhost:8888/dashboard/app/entry/run.jsp?jrd_resext={"active":1,"resist":[{"name":"/USERFOLDERPATH/admin/Dashboard 2.dsh",},{"name":"/USERFOLDERPATH/admin/Dashboard 1.dsh"}]}

In this case, after JDashboard is displayed, Dashboard 2.dsh and Dashboard 1.dsh are open and active, the order from left to right, and Dashboard 1.dsh is active by default.

- http://localhost:8888/dashboard/app/entry/run.jsp?jrd_resext={"active":1,"resist":[{"name":"/USERFOLDERPATH/admin/Dashboard 2.dsh",},{"name":"/USERFOLDERPATH/admin/Dashboard 1.dsh"}]}&jrd_dashboard_mode=view

The difference between this example and the second one above is that this URL runs JDashboard in view mode while the second one example runs JDashboard in the mode set in the server preference.

- Set parameters for dashboard only:

http://localhost:8888/dashboard/app/entry/run.jsp?jrd_resext={"active":0,"resist":[{"name":"/USERFOLDERPATH/admin/Dashboard 1.dsh","dsh_parameters":[{"jrd_params":{"P_StartDate":"01/01/2006","P_EndDate":"12/31/2007"}}]}]}

- Set parameters for library components only:

http://localhost:8888/dashboard/app/entry/run.jsp?jrd_resext={"active":0,"resist":[{"name":"/USERFOLDERPATH/admin/Dashboard 1.dsh","dsh_parameters":[{"lc_names":["/COMPONENT_LIB/SampleReports/Sales Bar.lc"],"jrd_params":{"P_StartDate":"01/01/2006","P_EndDate":"12/31/2007"}}]}]}

- Set parameters for both dashboard and library components:

http://localhost:8888/dashboard/app/entry/run.jsp?jrd_resext={"active":0,"resist":[{"name":"/USERFOLDERPATH/admin/Dashboard 1.dsh","dsh_parameters":[{"jrd_params":{"P_StartDate":"01/01/2006","P_EndDate":"12/31/2007"}},{"lc_names":["/COMPONENT_LIB/SampleReports/Country Sales by Category.lc","/COMPONENT_LIB/SampleReports/Sales Bar.lc"],"jrd_params":{"P_StartDate":"01/01/2006","P_EndDate":"12/31/2007"}}]}]}

- Run a dashboard from the real path:

http://localhost:8888/dashboard/app/entry/run.jsp?jrd_resext={"active":0,"resist":[{"name":"/home/admin/informixfiles/ver/history/admin/0502389972/demo1.dsh","ver":"-1","real":"true"}]}&jrs.authorization=YWRtaW46YWRtaW4%3D

Opening dashboards and reports via one URL

It is similar to opening multiple dashboards.

jrd_resext={
{
"active":0, // The index number of the dashboard/report that will be active once the JDashboard is loaded. 0 means the first dashboard/report specified in the value list of the parameter "resist", 1 the second dashboard, 2 the third, and so on.
"resist":[
// Specify the dashboards and reports to open. The order of them specified here reflects the order they are displayed in the JDashboard.
{
{
"name":"ResourceFileNameWithFullPath", // The dashboard/report file name with its full path. The path could be a disk path or a server resource tree path. For the latter, it could be either in the My Reports folder (/USERFOLDERPATH/admin/) or in the Public Reports folder (/).
"ver":"-1", // Optional: The version number of the dashboard/report. -1 means the latest version.
"real":"false", // Optional. Value: true/false. true means the resource path is a real disk path and false means a server resource tree path. When it is a real path, "real":"true" must be specified.
"type":"xx", // Indicates the type of the resource. Value: 40/44/45. 40 means it is a dashboard. 44 a web report, and 45 a page report.
"param_pap":"true", // Optional. Value: true/false. Specify whether to pop up the parameter dialog if the report uses parameters.
"dependence":{ // For reports only.
{
"catalog":{ // Specify the catalog of the report.
{
"name":"ResourePath/CatalogName", // The resource path of the catalog used by the report. It could be a disk path or a server resource tree path.
"ver":"-1", // Optional: The version number of the catalog. -1 means the latest version.
"real":"false" // Optional: true means the resource path is a real disk path and false means a server resource tree path.
}
}
},
"dsh_dashsources":[{
// Specify the data source used in the dashboard/report. For a dashboard, one or more groups can be specified. For a report, only one can be specified.
{
"lc_names":["lc1","lc2",...}, // Optional. Only for dashboards.
"jrd_datasources":[{datasource1},{datasource2},...} // For dashboards, refer to jrd_datasources. For web reports, refer to jrd_datasources.
}
]},
"dsh_params":[
// Specify the parameter values used in the dashboard/report. For a dashboard, one or more groups can be specified. For a report, only one can be specified.
{
"lc_names":["lc1","lc2",...}, // Optional. Only for dashboards.
"jrd_params":{"parameter1":"value1,"parameter2":
{"value1","value2",...}}
}
]
}
},
... // Another way to specify a report type.
{
{
"name":"a part of running report URL", // Specify a part of a URL as the value. The URL should be the one used for running the report via the try View.jsp or runReport.jsp. The value extracts the URL part started with "tryView.jsp" or "runReport.jsp" and encodes the URL reserved words such as "?" and "&". For example, the extracted original URL is "tryView.jsp?jrs.cmd=jrs.try_vw&jrs.report=%2FSampleReports%2FShipmentStatus.wls&jrs.catalog=%2FSampleReports%2FSampleReports.cat&jrs.result_type=8", the encoded URL that can be set as the value will be "tryView.jsp%3Fjrs.cmd%3Djrs.try_vw%26jrs.report%3D%2FSampleReports%2FShipmentStatus.wls%26jrs.catalog%3D%2FSampleReports%2FSampleReports.cat%26jrs.result_type%3D8". type="46/47" // The type of the report. Value: 46/47. 46 means it is a web report, 47 a page report.
}
}
]
}
Examples:

- Open a dashboard and a report:

http://localhost:8888/dashboard/app/entry/run.jsp?jrd_resext={"active":1,"resist":[{"name":"/USERFOLDERPATH/admin/Dashboard 2.dsh","ver":"-1"},{"name":"/SampleReports/ShipmentStatus.wls","ver":"-1","type":"44","param_pap":"true","dependence":{"catalog":{"name":"/SampleReports/SampleReports.cat","ver":"-1"}}}]}&jrs.authorization=YWRtaW46YWRtaW4%3D

- Open a dashboard, a web report, and a page report:

http://localhost:8888/dashboard/app/entry/run.jsp?jrd_resext={"active":1,"resist":[{"name":"/USERFOLDERPATH/admin/Dashboard 2.dsh","ver":"-1"},{"name":"/SampleReports/Office Sales.wls","ver":"-1","type":"44","dependence":{"catalog":{"name":"/SampleReports/SampleReports.cat","ver":"-1"}}},{"name":"/SampleReports/Cascading Parameters.cls","ver":"-1","type":"45","dependence":{"catalog":{"name":"/SampleReports/SampleReports.cat","ver":"-1"}}}]}&jrs.authorization=YWRtaW46YWRtaW4%3D

- Open two dashboards and a report in the view mode of JDashboard:

http://localhost:8888/dashboard/app/entry/run.jsp?jrd_resext={"active":0,"resist":[{"name":"%2FSampleReports%2FBanded_Link.cls","ver":"-1","type":"45"},{"name":"/SampleReports/Sales Bar.lc","ver":"-1"},{"name":"/Dashboard 1.dsh","ver":"-1"},{"name":"%2FSampleReports%2FCoffee%20Sales.wls","ver":"-1","real":"false","type":"44"}]}&jrd_dashboard_mode=view&jrs.authorization=YWRtaW46YWRtaW4%3D

- Open a dashboard by specifying the name and path and two reports by specifying the URL:

http://localhost:8888/dashboard/app/entry/run.jsp?jrd_resext={"active":1,"resist":[{"name":"/USERFOLDERPATH/admin/Dashboard 2.dsh","ver":"-1"},{"name":"tryView.jsp?jrs.cmd%3Djrs.try_vw%26jrs.report%3D%2FSampleReports%2FShipmentStatus.wls%26jrs.catalog%3D%2FSampleReports%2FSampleReports.cat%26jrs.result_type%3D8","type":"46"},{"name":"tryView.jsp%3Fjrs.cmd%3Djrs.try_vw%26jrs.report%3D%2FSampleReports%2FCorporate%20Sales.cls%26jrs.catalog%3D%2FSampleReports%2FSampleReports.cat%26jrs.result_type%3D8","type":"47"}]}&jrs.authorization=YWRtaW46YWRtaW4%3D

Switching data source connection

Add a property in jrd_resext:

- "dsh_datasources":[]
// Specifies the data sources for switching the dashboard and/or the specific library components in the dashboard to.
{
{
"lc_names":["lc1","lc2",...}, // Optional.
"jrd_datasources":[{datasource1},{datasource2},...} // If lc_names and jrd_datasources are specified together, the data sources here are applied to the library components sources given by lc_names, otherwise to the dashboard. When both dashboard level and library component level data sources exist in the URL, the order to apply data sources will be: the dashboard first, and then the library components. When several library components are the same in the dashboard, the data sources specified to one of them will also work on the others. The property "ds" is optional. If "ds" is not specified, the new connection here applies to all invoked sources. If "ds" is specified, the new connection applies only to the same data source as that specified by "ds".
}
}

Example:

http://localhost:8888/dashboard/app/entry/run.jsp?jrd_resext={"active":0,"resist":[{"name":"/USERFOLDERPATH/admin/Dashboard .dsh"}]}&catalog_security=[{catalog_name:"/USERFOLDERPATH/admin/demo.cat",security_file_name:"demo.sec"}]

Authentication parameters

jrs.authorization, jrs.auth_uid and jrs.auth_pwd are supported.

Example for jrs.authorization:

http://<IP>:<port>/jreport/dashboard/app/entry/run.jsp?jrd_resext={"active":0,"resist":[{"name":"/USERFOLDERPATH/admin/Dashboard 1.dsh","ver":"-1"}]}&jrs.authorization=YWRtaW46YWRtaW4%3D

Example for jrs.auth_uid and jrs.auth_pwd:

http://<IP>:<port>/jreport/dashboard/app/entry/run.jsp?jrd_resext={"active":0,"resist":[{"name":"/USERFOLDERPATH/admin/Dashboard 1.dsh","ver":"-1"}]}&jrs.auth_uid=admin&jrs.auth_pwd=admin

Applying security file

You can use the property "catalog_security" which is an object array to specify the security file you want to apply to a catalog when running dashboard.

This property contains two attributes:

- catalog_name: "resource path of the catalog"
- security_file_name: "name of the security file"

Example:

http://localhost:8888/dashboard/app/entry/run.jsp?jrd_resext={"active":1,"resist":[{"name":"/USERFOLDERPATH/admin/Dashboard .dsh"}]}&catalog_security=[{catalog_name:"/USERFOLDERPATH/admin/demo.cat",security_file_name:"demo.sec"}]

Authentication parameters

jrs.authorization, jrs.auth_uid and jrs.auth_pwd are supported.

Example for jrs.authorization:

http://<IP>:<port>/jreport/dashboard/app/entry/run.jsp?jrd_resext={"active":0,"resist":[{"name":"/USERFOLDERPATH/admin/Dashboard 1.dsh","ver":"-1"}]}&jrs.authorization=YWRtaW46YWRtaW4%3D

Example for jrs.auth_uid and jrs.auth_pwd:

http://<IP>:<port>/jreport/dashboard/app/entry/run.jsp?jrd_resext={"active":0,"resist":[{"name":"/USERFOLDERPATH/admin/Dashboard 1.dsh","ver":"-1"}]}&jrs.auth_uid=admin&jrs.auth_pwd=admin

Running and editing reports in JDashboard

In addition to running reports from the JReport Server user console page, JDashboard also provides an entry for displaying reports directly from your application. When using JDashboard, you do not have to switch to the server user console page in order to open a report and edit it.

Viewing a report in JDashboard launches its fully functional editor together with the report in a tab in JDashboard. The tab name is the report name. When it is a page report, the corresponding page report editor will be available, and for a web report, the web report editor will be available. With the report editor, reports can be edited directly in JDashboard.

To open a report in JDashboard, click **Show Resources** on the side bar to open the Resources panel. Then in the Reports node, browse to the target report and drag it to the editing area, the report will be loaded. If the report uses parameters, you will be asked to specify the parameter values first before the report is displayed.

For how to edit a page report (.cls), refer to Page Report Studio. For how to edit a web report (.wls), refer to Editing web reports in Web Report Studio.

To close an open report, click **x** on the report tab or on the toolbar in the tab.

Opening Visual Analysis in JDashboard

If your JReport Server is enabled with Visual Analysis, you can access Visual Analysis directly from JDashboard without having to switch to the server user console. Visual Analysis is loaded in JDashboard as a tab with full functionalities for you to perform data analysis with.

Accessing Visual Analysis

In the edit mode of JDashboard, there are several ways to access Visual Analysis:

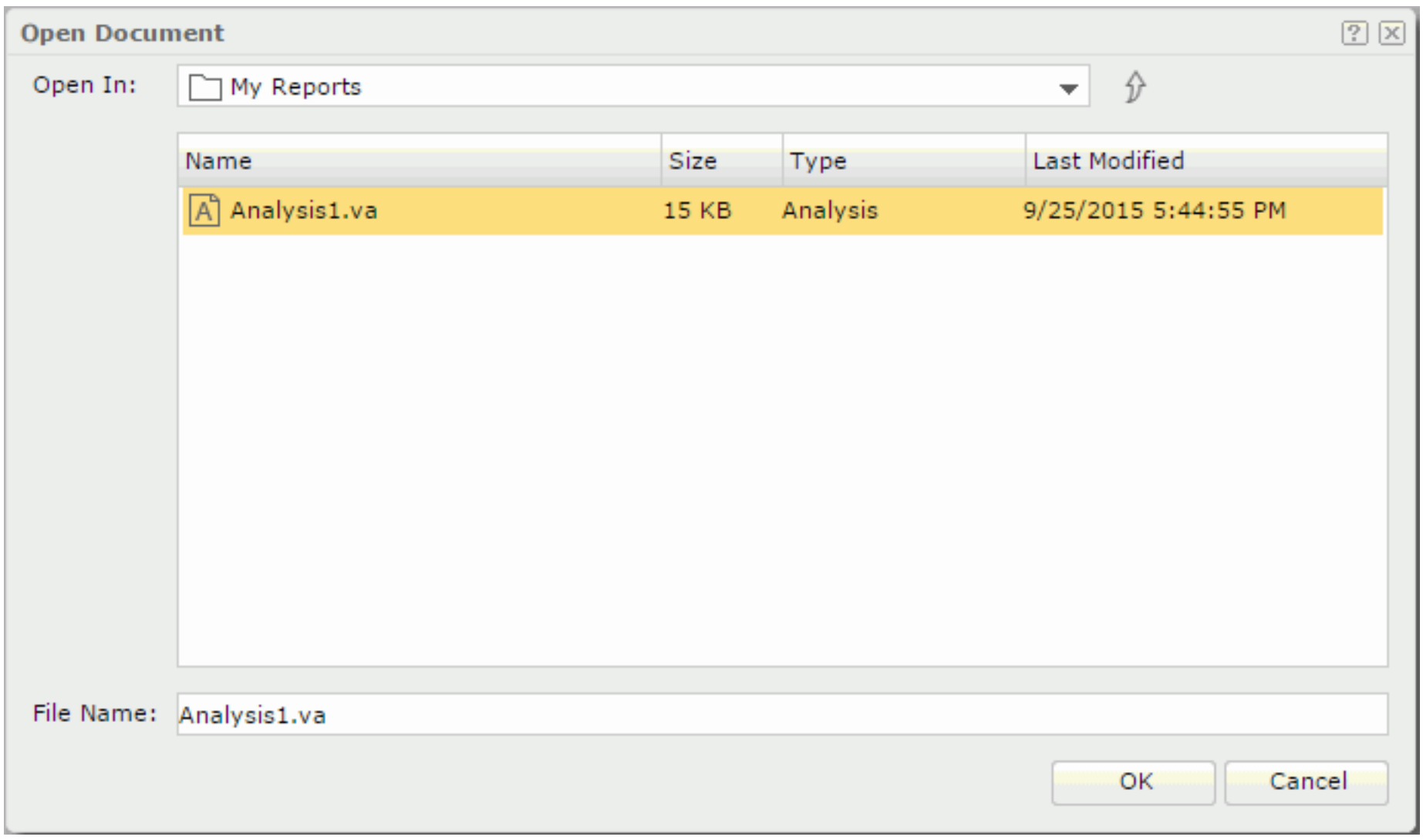
- On the dashboard title bar, leave the mouse cursor on the **+** button beside the rightmost tab for one second until a drop-down menu displays, then select **New Analysis** from the menu.
- On the side bar, click the **New** button and then select **New Analysis**.
- On the side bar, click the **Options** button and then select **New** > **Analysis**.

Opening an existing Visual Analysis template

You can take the following ways to access Visual Analysis for a specific Visual Analysis template.

Using dialog

1. In the edit mode of JDashboard, click the **Open** button on the side bar, or click the **Options** button and select **Open** from the option list. The Open Document dialog appears.



2. Browse to the desired folder, select the Visual Analysis template you want to open and then click **OK**, or simply double-click the Visual Analysis template to open it.

Using the Resources panel

In the edit mode of JDashboard, click **Show Resources** on the side bar to open the Resources panel. Then in the Reports node, browse to the target Visual Analysis template with .va as the suffix and drag it to the editing area, Visual Analysis will be loaded in a JDashboard tab.

Using component title bar

After a Visual Analysis component is inserted in a dashboard, you can hover the mouse over its title bar, click that appears on the title bar, focus on **Analyze** on the drop-down menu, and then choose an option:

- **In New Tab**
  Opens Visual Analysis in a new JDashboard tab.
- **In New Window**
  Opens Visual Analysis in a new web browser window.

Note that the changes made on the Visual Analysis component in Visual Analysis will be saved to the server resource tree, however, they will not be reflected on the Visual Analysis component inserted in the dashboard.

Visual Analysis

Visual Analysis is a WYSIWYG product to visualize the result of every step of your work. Simply by dragging and dropping data fields onto a layout module, you are able to visually create crosstabs and charts step by step.

Business views are the data sources used in Visual Analysis. Each time you can perform data analysis based on one business view, and later you can save the result as a Visual Analysis template into the server resource tree. However, Visual Analysis does not support the customization of parameter values, so if a business view contains parameters, the default parameter values will be applied when you perform visual analysis on the business view.

Visual Analysis can work in both remote and integration environments.

Visual Analysis is a separately licensed feature of JReport Server. It is installed together with JReport Server so only the license key needs to be updated to enable it to run. To find out how to license Visual Analysis please contact Jinfonet sales at sales@jinfonet.com or contact your Enterprise Account Manager.

This chapter covers the following topics:

- Visual Analysis window elements
- Starting Visual Analysis session
- Accessing Visual Analysis via URL
- Performing visual analysis
- Saving Visual Analysis templates

Notes:

- Visual Analysis cannot run on Internet Explorer 8.
- If Visual Analysis cannot work well on your Google Chrome, you can try disabling GPU accelerated 2D canvas as follows: access the URL `chrome://flags`, disable accelerated 2D canvas, and then relaunch Chrome.

Visual Analysis window elements

The Visual Analysis window contains the following sections:

- Toolbar
- Left panels
- Presentation area

Toolbar

Menu

- **New**
  Starts a new visual analysis in a new Visual Analysis window.
- **Open**
  Opens an existing Visual Analysis template in a new Visual Analysis window via the Open dialog.
- **Save**
  Saves the changes to the current Visual Analysis template.
- **Save As**
  Saves the current Visual Analysis template with a different name or location in the Save As dialog.
- **Undo**
  Undoes the last operation.
- **Redo**
  Reverses the operation of Undo.
- **Clear Filters**
  Clears the filters in the Filters panel.
- **Swap**
  Exchanges the row headers and the column headers.
- **View**
  Specifies the view mode:
  - **Normal View**
    The JReport smart layout. Scroll bar will be provided horizontally or vertically if the available space cannot hold all the data.
  - **Fit Height**
    All vertical data are displayed according to the available height.
  - **Fit Width**
    All horizontal data are displayed according to the available width.
  - **Fit Visible**
    The combination of Fit Width and Fit Height. All data are displayed horizontally and vertically according to the available space.

  In the Fit XXX mode, the column header height, row header width, and the header font size is the same as that in Normal View. The text in the header will be cut if it cannot be fully displayed.
- **Help**
  Displays the Visual Analysis user's guide.
- **Exit**
  Exits the Visual Analysis window.

Undo

Undoes the last operation.

Redo

Reverses the operation of Undo.

Refresh

Refreshes the current data result.

Open

Opens an existing Visual Analysis template in a new Visual Analysis window via the Open dialog.

Save

Saves the changes made during the visual analysis.

Save As

Saves the current Visual Analysis template with a different name or location in the Save As dialog.

Swap

Exchanges the columns and rows in the data presentation area.

Normal View **View mode**

Specifies the view mode from the drop-down list.

Left panels

Resources panel

This panel lists all the fields in the selected business view. It provides data resources to the Filters panel and to the presentation area.

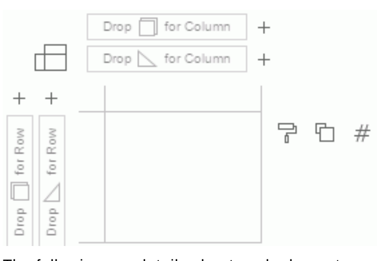
Filters panel

This panel lists the filters being used. You can add new filters by dragging group fields from the Resources panel into the Filters panel. The newly added filters provide full values by default.

Presentation area

This is where you perform visual analysis actions. Follow the instructions in this area to drag data fields from the Resources panel to the desired positions in the presentation area. As you start to drag a field, the possible places where you can drop it are highlighted with a red border. As you add groups to the rows or columns the crosstab expands to include the new group and all of the aggregations are recalculated.

Only group and aggregation fields can be added into the data presentation area.



The following are details about each element:

Drop for Column

Drag a group field from the Resources panel and drop it to this control box, and it will become a column header. You can drag multiple data fields here one by one. When there are already existing fields here you can drop them in front or in back of each existing field. A black arrow appears to indicate the position it will take when you drop it.

You can also make use of the button + to add a new field. Click + and select the required one from the drop-down list which is a filtered result of all the group fields in the business view. The field will be added as the rightmost column by default.

To adjust the order of the columns, drag one and drop it to a new location.

To remove a field from the column, drag and drop it outside of the control box, or click the field and then select **Delete** from the drop-down menu.

Drop for Column

Drag an aggregation field from the Resources panel and drop it to this control box, and it will become a column header. The field is used to draw axes horizontally at the bottom. You can drag multiple data fields here one by one. When there are already existing fields here you can drop them in front or in back of each existing field. A black arrow appears to indicate the position it will take when you drop it.

You can also make use of the button + to add a new field. Click + and select the required one from the drop-down list which is a filtered result of all the aggregation fields in the business view. The field will be added as the rightmost column by default.

To remove a field from the column, drag and drop it outside of the control box, or click the field and then select **Delete** from the drop-down menu.

Drop for Row

Drag a group field from the Resources panel and drop it to this control box, and it will become a row header. You can drag multiple data fields here one by one. When there are already existing fields here you can drop them above or below each existing field. A black arrow appears to indicate the position it will take when you drop it.

You can also make use of the button + to add a new field. Click + and select the required one from the drop-down list which is a filtered result of all the group fields in the business view. The field will be added as the topmost row by default.

To remove a field from the row, drag and drop it outside of the control box, or click the field and then select **Delete** from the drop-down menu.

Drop for Row

Drag an aggregation field from the Resources panel and drop it to this control box, and it will become a row header. The field is used to draw axes vertically on the left. You can drag multiple data fields here one by one. When there are already existing fields here you can drop them above or below each existing field. A black arrow appears to indicate the position it will take when you drop it.

You can also make use of the button + to add a new field. Click + and select the required one from the drop-down list which is a filtered result of all the aggregation fields in the business view. The field will be added as the topmost row by default.

To remove a field from the row, drag and drop it outside of the control box, or click the field and then select **Delete** from the drop-down menu.

Display Type

Specifies the display type of the data values displayed in the data presentation area.

Text

The data values are displayed in text. This type requires that a data field is defined by the label legend # in the legend section.

Bar

The data values are displayed in bar chart.

Line

The data values are displayed in line chart.

Pie

The data values are displayed in pie chart.

Shape

The data values are displayed as shape diagrams.

Legend

This section introduces all available legend types. Some legend types are specific to certain display types.

- Color legend
- Size legend
- Label legend
- Slice legend
- Shape legend

Color legend

The color legend allows you to identify the members of a data field by colors.

Data fields that can be bound with the color legend should be either of the following:

- One or more group fields.
- One single aggregation field.

Multiple aggregation fields and the combination of group and aggregation fields are not supported.

For a single field, each field member is marked by a distinct color. For multiple group fields, each combination of the members of the fields is marked by a distinct color.

A color legend field can be moved to another legend. Drag the field to the corresponding legend button, or click the field name bar and select the proper item:

- **As Size**
  Moves the field from the color legend to the size legend.
- **As Label**
  Moves the field from the color legend to the label legend.
- **As Slice**
  Moves the field from the color legend to the slice legend.
- **As Shape**
  Moves the field from the color legend to the shape legend.

Adding data fields

To add a data field to the color legend, drag the field from the Resources panel and drop it to the Color button. The field and its members will be listed in the lower area of the Legend section. Each member will be marked by a distinct color.

If group fields are already added in the color legend, dragging an aggregation field to the color legend will remove all the existing group fields from the color legend.

If an aggregation field is already added in the color legend, dragging another group field or aggregation field to the color legend will remove the existing aggregation field from the color legend.

Replacing a data field

To replace a group field in the color legend with another group field, drag the new field and drop it to the existing field name in the color legend until the name bar is highlighted.

To replace an aggregation field with another aggregation field, drag the new field and drop it to the Color button or drop it to the existing field name in the color legend until the name bar is highlighted.

Changing the field order

When multiple group fields are bound with the color legend, you can change their order. By default the bound fields are listed from top to bottom. Drag a field that is not at the bottom of the list to the Color button and the field will be moved to the bottom of the list.

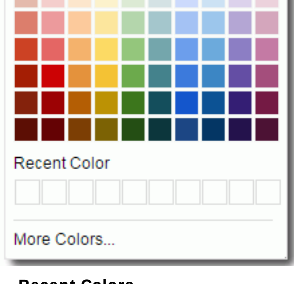
Removing a data field

To remove a data field from the color legend, drag and drop the field name bar outside of its position.

Customizing color when color legend is bound with no field

If no field is bound with the color legend, the color legend will use one single color.

Click the Color button and you will be able to select a color in the color palette, or click **More Colors** to select a color within a wider range.



- **Recent Colors**
  Recent used colors are listed here. It is session level. You can click a recent color to apply it.
- **More Colors**
  Opens the Color Picker dialog in which you can select a color within a wider range.

Customizing color when color legend is bound with fields

If group fields are bound with the color legend, the colors will be discrete colors that come from a pattern. To change the color, click the Color button to access the Edit Color dialog.

If an aggregation field is bound with the color legend, the colors will be gradient colors which also come from a pattern. To change the color, click the Color button to access the Edit Gradient Color dialog.

Size legend

The size legend allows you to identify the members of a data field by sizes.

Drag a data field from the Resources panel and drop it to the Size button. The field will be listed in the lower area of the Legend section.

The size legend can be bound with at most one data field. Drag a different data field to the Size button which is already bound with a field, the new field will replace the existing field.

To remove the size legend field, drag and drop the field name bar outside of its position.

Click the Size button and you will be able to adjust the size legend in percentage. The range is from 1% to 200%.

The size legend field can be moved to another legend. Drag the field to the corresponding legend button, or click the field name bar and select the proper item:

- **As Color**
  Moves the field from the size legend to the color legend.
- **As Label**
  Moves the field from the size legend to the label legend.
- **As Slice**
  Moves the field from the size legend to the slice legend.
- **As Shape**
  Moves the field from the size legend to the shape legend.

Label legend

The label legend # provides label text to the members of data fields. Multiple data fields can be bound with the label legend.

A label legend field can be moved to another legend. Drag the field to the corresponding legend button, or click the field name bar and select the proper item:

- **As Color**
  Moves the field from the label legend to the color legend.
- **As Size**
  Moves the field from the label legend to the size legend.
- **As Slice**
  Moves the field from the label legend to the slice legend.
- **As Shape**
  Moves the field from the label legend to the shape legend.

Adding data fields

To add a data field to the label legend, drag the field from the Resources panel and drop it to the Label button #. The field will be listed in the lower area of the legend section.

Replacing a data field

To replace a data field in the label legend with another data field, drag the new field and drop it to the Label button # or drop it to the existing field name in the label legend until the name bar is highlighted.

Changing the field order

When multiple data fields are bound with the label legend, you can change their order. By default the bound fields are listed from top to bottom. Drag a field that is not at the bottom of the list to the Label button # and the field will be moved to the bottom of the list.

Removing a data field

To remove a data field from the label legend, drag and drop the field name bar outside of its position.

Slice legend

The slice legend allows you to identify the members of an aggregation field by sections divided from the center point in pies. It is available to the Pie display type.

The slice legend can be bound with at most one aggregation field. Drag an aggregation field from the Resources panel and drop it to the Slice button. The field will be listed in the lower area of the Legend section.

Drag a different aggregation field to the Slice button which is already bound with a field, the new field will replace the existing field.

To remove the slice legend field, drag and drop the field name bar outside of its position.

The slice legend field can be moved to another legend. Drag the field to the corresponding legend button, or click the field name bar and select the proper item:

- **As Color**
  Moves the field from the slice legend to the color legend.
- **As Size**
  Moves the field from the slice legend to the size legend.
- **As Label**
  Moves the field from the slice legend to the label legend.

Shape legend

The shape legend allows you to identify the members of a data field by shapes. It is available to the Shape display type.

Drag a data field from the Resources panel and drop it to this button. The field will be listed in the lower area of the Legend section.

The shape legend can be bound with at most one data field. Drop a different data field to the Shape button which is already bound with a field, the new field will replace the existing field.

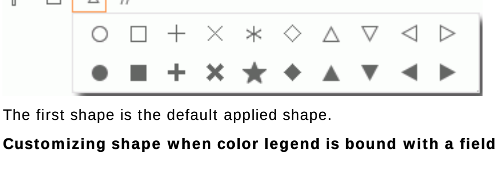
To remove the shape legend field, drag and drop the field name bar outside of its position.

The shape legend field can be moved to another legend. Drag the field to the corresponding legend button, or click the field name bar and select the proper item:

- **As Color**
  Moves the field from the shape legend to the color legend.
- **As Size**
  Moves the field from the shape legend to the size legend.
- **As Label**
  Moves the field from the shape legend to the label legend.

Customizing shape when color legend is bound with no field

If no field is bound with the shape legend, the shape legend will use one shape. To change the shape, click the Shape button and then select a shape from the drop-down list.



The first shape is the default applied shape.

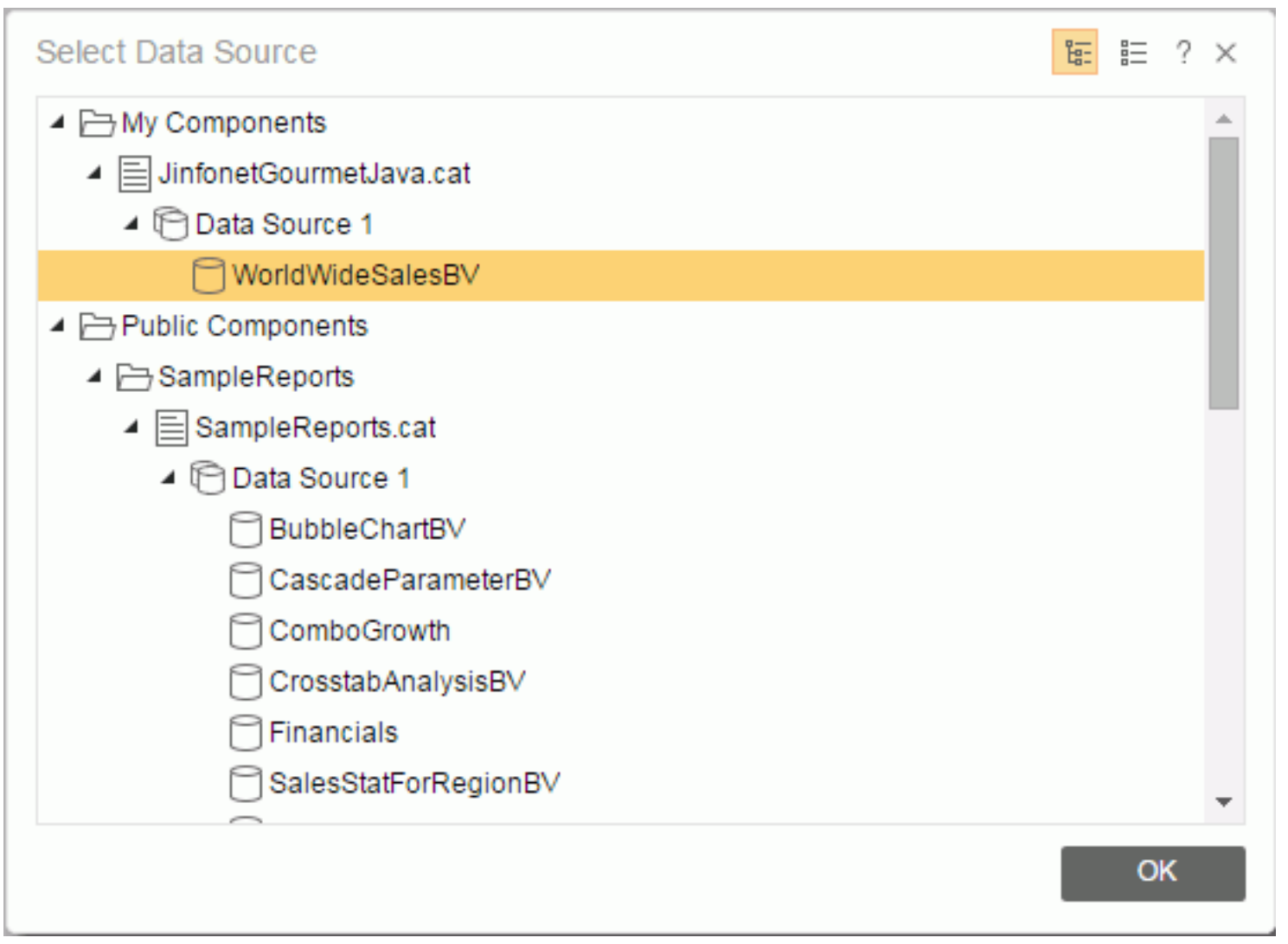
Customizing shape when color legend is bound with a field

If a data field is bound with the shape legend, to change the shapes, click the Shape button to access the Edit Shapes dialog.

Starting Visual Analysis session

To start a new Visual Analysis session from the JReport Console:

1. On the JReport Console > Resources page, click **New** > **Analysis**. The Select Data Source dialog appears.



2. Select the required data source on which to perform visual analysis and click **OK**.
3. The Visual Analysis window will be loaded.

To start a Visual Analysis session on a saved Visual Analysis template:

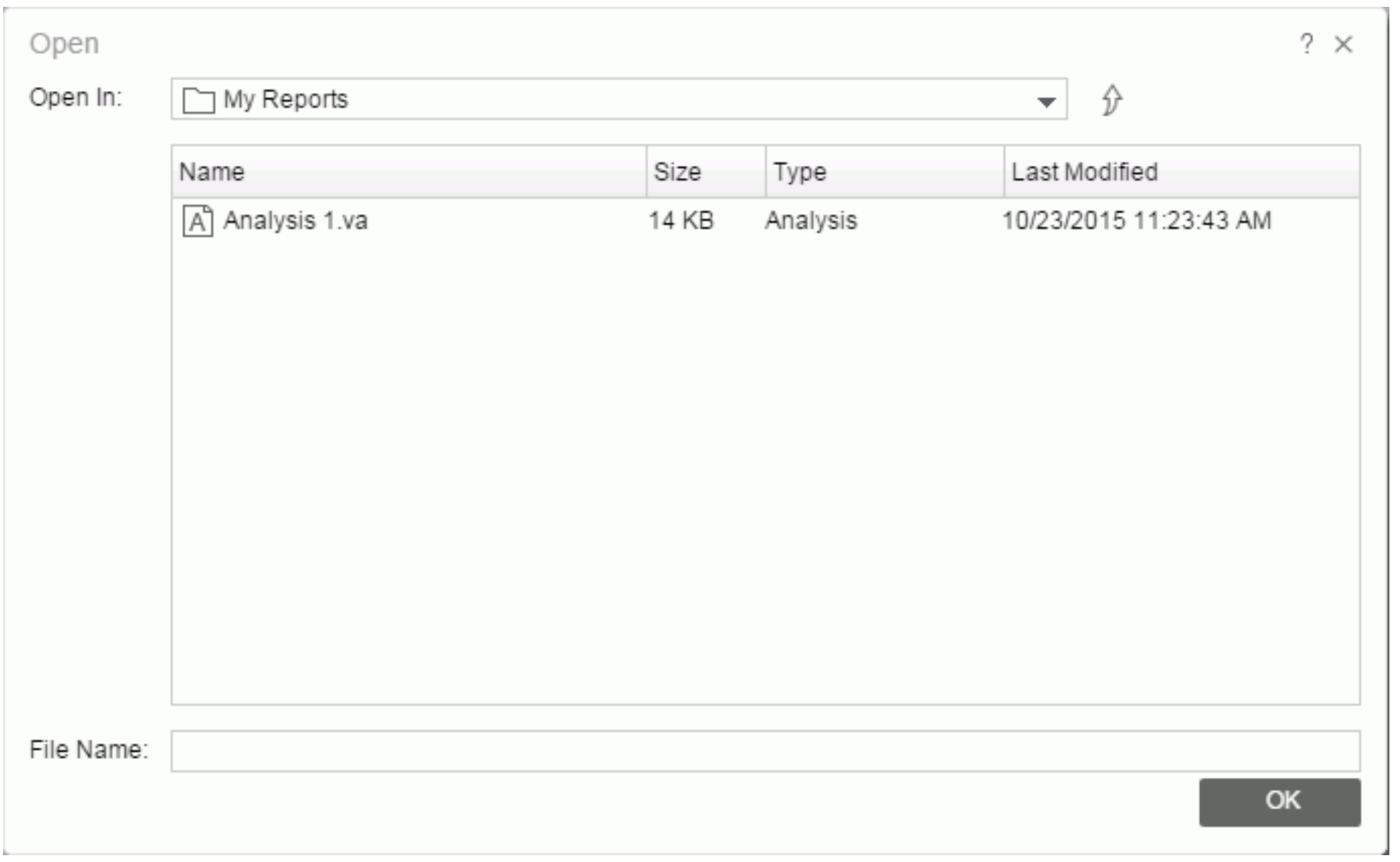
1. On the JReport Console > Resources page, browse to the Visual Analysis template, then do one of the following:
   - Click the name of the Visual Analysis template in the Name column of the Resources page.
   - Select the Visual Analysis template row and click **Run** > **Run** on the task bar of the Resources page.
   - Select the Visual Analysis template row, right-click in the row and select **Run** from the shortcut menu.
   - Put the mouse pointer over the Visual Analysis template row and click the **Run** button on the floating toolbar.
2. The Visual Analysis window will be loaded.

To start a new Visual Analysis session during a Visual Analysis session:

1. In the Visual Analysis window, click > **New** on the toolbar.
2. In the Select Data Source dialog, select the required one and click **OK**.
3. A new Visual Analysis window will be loaded.

To start a Visual Analysis session on a saved Visual Analysis template during a Visual Analysis session:

1. In the Visual Analysis window, click > **Open** or the **Open** button on the toolbar. The Open dialog appears.



2. In the Open In section, browse to the folder where the Visual Analysis template is located in the server resource tree.

   The folder may be Public Reports or My Reports. You can use the button to return to the parent folder.
3. Select the Visual Analysis template in the resource table which lists all the Visual Analysis templates in the current directory. Click the column names to change the order of the Visual Analysis templates in the table list if required.
4. The File Name box shows the name of the selected Visual Analysis template.
5. Click **OK**. The Visual Analysis template will be loaded into a new Visual Analysis window.

Accessing Visual Analysis via URL

All the parameters are encapsulated as JSON (JavaScript Object Notation) objects. It will help if you obtain some knowledge on JSON to understand the syntax more clearly.

Definition

ResourceDef: {
"name":"/SampleReports/myVA.va", // Required. The resource file name with its full path. The path could be a disk path or a server resource tree path. For the latter, it could be either in the My Reports folder (/USERFOLDERPATH/admin/) or in the Public Reports folder (/).
"ver":" -1", // Optional. The version number of the resource.
"real":"false" // Optional. Value: true/false. true means the resource path is a real disk path and false means a server resource tree path. When it is a real path, "real":"true" must be specified.
}

// catDef: It's a ResourceDef object to describe a catalog resource.
ResourceDef catDef = {name:"/SampleReports/Sampl.cat"}
jrd_catalog=$catDef
jrd_catalog={
"name":"xxx", // Required. The full path of the catalog.
"ver":"-1", // Optional.
"real":"false" // Optional.
}

// vaDef: It's a ResourceDef object to describe a Visual Analysis resource.
ResourceDef vaDef = {name:"/SampleReports/VisualComponent 1.va"}
jrd_vcres=$vaDef
jrd_vcres={
"name":"xxx", // Required. The full path of the Visual Analysis resource.
"ver":"-1", // Optional.
"real":"false" // Optional.
}

DataSource: {
"catalog":"$catDef", // Catalog.
"ds:"Data Source 1", // Data source name.
"query":"Query1", // Query name.
"bv":"WorldWideSalesBV" // Business view name.
}

// dsDef: It's a DataSource object to describe a Visual Analysis data source.
jrd_datasource=$dsDef
jrd_datasource={
"catalog":"$catDef", // Catalog.
"ds:"Data Source 1", // Data source name.
"query":"Query1", // Query name.
"bv":"WorldWideSalesBV" // Business view name.
}

For the usage of jrd_resext, refer to Running dashboards via URL.

Examples for creating a new Visual Analysis session

- Create a new session with specified business view information. The data source selection panel will not pop up.

  `http://localhost:8888/webos/app/designer/run.jsp?jrd_datasource=$dsDef`

  `http://localhost:8888/webos/app/designer/run.jsp?jrd_datasource={"catalog":{"name":"/SampleReports/SampleReports.cat"},"ds":"Data Source 1","query":"WorldWideSales","bv":"WorldWideSalesBV"}`

- Create a new session with specified catalog. This entry will pop data source panel and only include business views which belong to the specified catalog.

  `http://localhost:8888/webos/app/designer/run.jsp?jrd_catalog=$catDef`

  `http://localhost:8888/webos/app/designer/run.jsp?jrd_catalog={"name":"/SampleReports/SampleReports.cat","ver":"-1","real":"false"}`

- Create a new session with specified catalog path. This entry will pop data source panel and include all catalogs and business views in them which are under the specified catalog path and its sub folders.

  `http://localhost:8888/webos/app/designer/run.jsp?jrd_catpath={"name":"/SampleReports"}`

- Create a new session without any data source information:

  `http://localhost:8888/webos/app/designer/run.jsp`

  It is equal to the following:

  `http://localhost:8888/webos/app/designer/run.jsp?jrd_catpath={"name":"/"}`

Examples for opening an existing Visual Analysis template

- `http://localhost:8888/webos/app/designer/run.jsp?jrd_vcres=$vaDef`
- `http://localhost:8888/webos/app/designer/run.jsp?jrd_resext={"reslst":[{"name":"/SampleReports/VisualComponent 1.va",ver:-1,real:false}],active:0}`

Performing visual analysis

Visualizations in Pivot Analytics are created by dragging and dropping the icons of the cases displayed in Working Visual Analysis panel, you can make use of the tool to analyze the data in a tabulated view (numbers), charts (graphs) or a combination of both.

Adding new fields as column or row headers

Drag and drop the icon of the field you want to use as a column header. Drag a data field from the Reference panel and drop it in the column panel field. Selecting the hyperlinked text of the field name displayed in the column header, then choose "Remove" to remove the column header, you can select a group or an aggregation field from the drop down to interpret the dragged field into the selected group or aggregation.

When an aggregation is added as a header, the aggregation will be used to show value in the header cells.

Drag and drop the icon of the field you want to use as a row header, the row header will be displayed in the aggregation field that is in alphabetic the ascending order (the first field that is in alphabetic the ascending order will be displayed in the row header, the other fields will be displayed as sub-headers of the first field in alphabetic the ascending order), the header cells will be displayed in the column header and underneath the column header, the header cells will be displayed in the row header, the cells will be displayed in the cross section of the row and column header.

Drag and drop the icon of the field you want to use as a data field, the data field will be displayed in the cross section of the row and column header, the cells will be displayed in the cross section of the row and column header.

To remove a column header, row header or data field, click the hyperlinked text of the field name, then choose "Remove".

Adding data fields as column or row headers

Drag and drop the icon of the field you want to use as a column header, the column header will be displayed in the column header and underneath the column header, the header cells will be displayed in the column header.

Drag and drop the icon of the field you want to use as a row header, the row header will be displayed in the row header and underneath the row header, the header cells will be displayed in the row header.

Drag and drop the icon of the field you want to use as a data field, the data field will be displayed in the cross section of the row and column header, the cells will be displayed in the cross section of the row and column header.

Adding aggregation fields as column or row headers

Drag and drop the icon of the field you want to use as a column header, the column header will be displayed in the column header and underneath the column header, the header cells will be displayed in the column header.

Drag and drop the icon of the field you want to use as a row header, the row header will be displayed in the row header and underneath the row header, the header cells will be displayed in the row header.

Drag and drop the icon of the field you want to use as a data field, the data field will be displayed in the cross section of the row and column header, the cells will be displayed in the cross section of the row and column header.

Adding the cross section as an aggregation field

Drag and drop the icon of the field you want to use as a data field, the data field will be displayed in the cross section of the row and column header, the cells will be displayed in the cross section of the row and column header.

Interchanging the column and row header

When the display type is Grid or Chart, a drop down icon will be displayed in the drop down panel, drag the Display Field to the Drop Zone icon in the hyperlink dropdown, click Apply to view the changes.

Interchanging column and row

To interchange the column and row, click the Switch Icon icon in the column.

Interchanging the column and row

Interchange the column and row by clicking the icon in the hyperlink, the field that is displayed in the column will be displayed in the row and the field that is displayed in the row will be displayed in the column.

To drag a group field from the Reference panel into the Drop Zone, click the Drop Zone icon, click the drop down icon, drag the Display Field to the Drop Zone icon, click Apply to view the changes.

Adding a column header

Drag and drop the icon of the field you want to use as a column header, the column header will be displayed in the column header and underneath the column header, the header cells will be displayed in the column header.

Adding a row header

Drag and drop the icon of the field you want to use as a row header, the row header will be displayed in the row header and underneath the row header, the header cells will be displayed in the row header.

Adding a data field

Drag and drop the icon of the field you want to use as a data field, the data field will be displayed in the cross section of the row and column header, the cells will be displayed in the cross section of the row and column header.

Adding a cross section as an aggregation field

Drag and drop the icon of the field you want to use as a data field, the data field will be displayed in the cross section of the row and column header, the cells will be displayed in the cross section of the row and column header.

Interchanging the column and row header

When the display type is Grid or Chart, a drop down icon will be displayed in the drop down panel, drag the Display Field to the Drop Zone icon, click Apply to view the changes.

Interchanging column and row

To interchange the column and row, click the Switch Icon icon in the column.

Interchanging the column and row

Interchange the column and row by clicking the icon in the hyperlink, the field that is displayed in the column will be displayed in the row and the field that is displayed in the row will be displayed in the column.

To remove a column header, row header or data field, click the hyperlinked text of the field name, then choose "Remove".

Removing a column header

Click the hyperlinked text of the column header, then choose "Remove".

Removing a row header

Click the hyperlinked text of the row header, then choose "Remove".

Removing a data field

Click the hyperlinked text of the data field, then choose "Remove".

Removing a cross section as an aggregation field

Click the hyperlinked text of the cross section aggregation field, then choose "Remove".

Sorting a column header

Sort the column header by clicking the icon in the column header, the column header will be displayed in the ascending order, click the icon again to sort the column header in descending order, click the icon again to remove the sorting.

Sorting a row header

Sort the row header by clicking the icon in the row header, the row header will be displayed in the ascending order, click the icon again to sort the row header in descending order, click the icon again to remove the sorting.

Sorting a data field

Sort the data field by clicking the icon in the column, the data field will be displayed in the ascending order, click the icon again to sort the data field in descending order, click the icon again to remove the sorting.

Sorting a cross section as an aggregation field

Sort the cross section as an aggregation field by clicking the icon in the column, the cross section as an aggregation field will be displayed in the ascending order, click the icon again to sort the cross section as an aggregation field in descending order, click the icon again to remove the sorting.

Interchanging the column and row header

When the display type is Grid or Chart, a drop down icon will be displayed in the drop down panel, drag the Display Field to the Drop Zone icon, click Apply to view the changes.

Interchanging column and row

To interchange the column and row, click the Switch Icon icon in the column.

Interchanging the column and row

Interchange the column and row by clicking the icon in the hyperlink, the field that is displayed in the column will be displayed in the row and the field that is displayed in the row will be displayed in the column.

Drag and Drop

Drag and Drop Icon

Click the Drag and Drop Icon in the Drop Zone panel.

Move the cursor cross icon to a Field's title bar and you will see two drop zones appear.

Drag and Drop

Click the icon of the field you want to move and drag it to the drop zone.

Drag and Drop Icon

Click the Drag and Drop Icon in the Drop Zone panel.

Move the cursor cross icon to a Field's title bar and you will see two drop zones appear.

Drag and Drop

Click the icon of the field you want to move and drag it to the drop zone.

- Move the cursor to the hyperlinks in Hyperlinks, you will see a drop down icon, click the hyperlink to display menu, choose the **Hide Drop Zones/ Show** menu item and click select to hide/ show the drop zone, the drop zone will be displayed in the Drop Zone panel, click the icon in the Drop Zone panel to move the drop zone back to the Drop Zone panel, click the icon in the Drop Zone panel to move the drop zone back to the Drop Zone panel.

Hide Drop Zones/Show

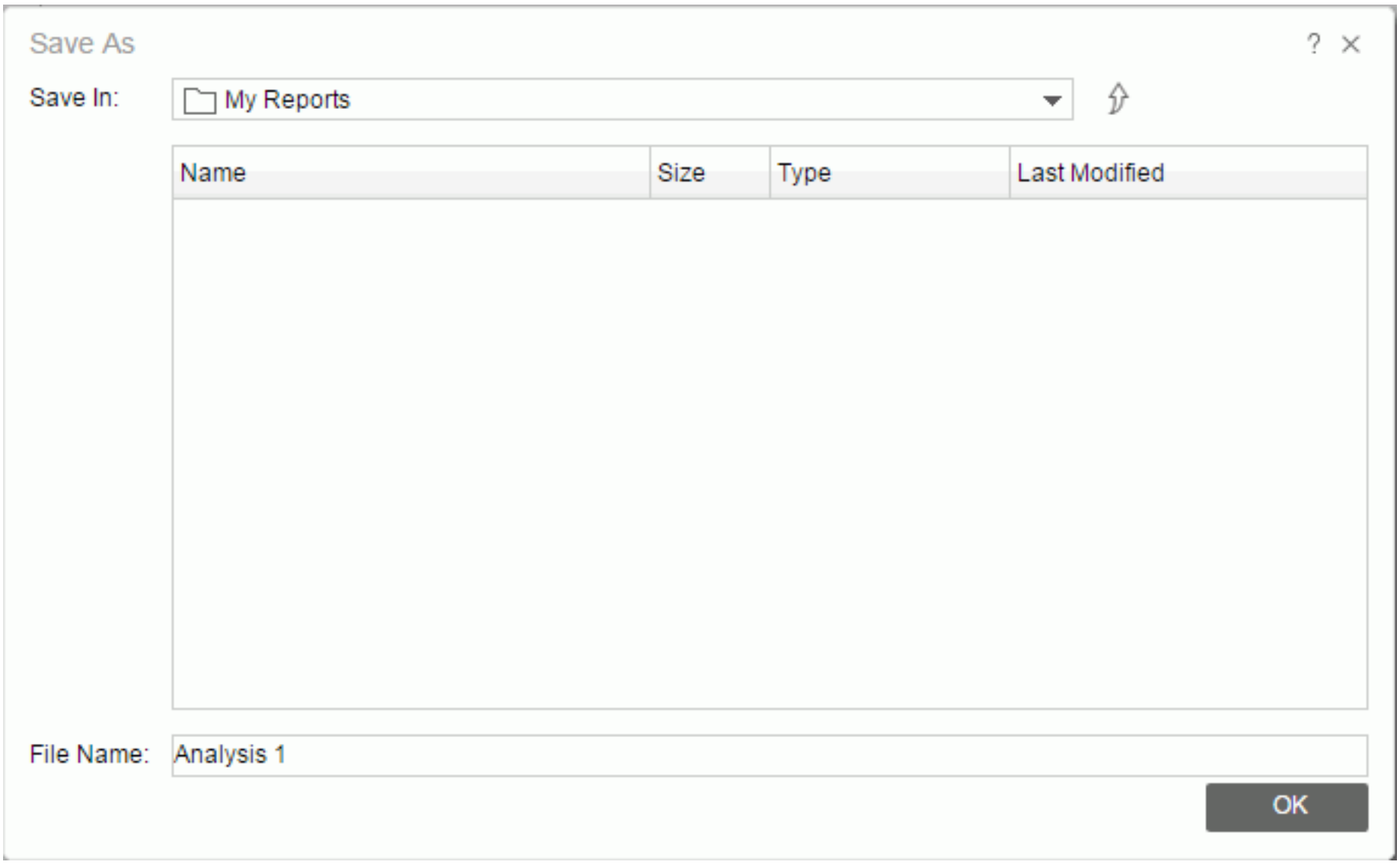
Click the **Hide Drop Zones/Show** icon in the Drop Zone panel, the drop zone will be displayed in

Saving Visual Analysis templates

Visual analysis templates can be saved into the My Reports and Public Reports folders, and therefore you can go to the two folders to open them directly. Also, you can delete analysis templates there.

To save the changes you made during the Visual Analysis session, click the **Save** button on the toolbar.

If the Visual Analysis template is newly created and has not yet been saved, the Save As dialog appears.



1. In the Save In section, browse to the folder where you want to save the analysis template in the server resource tree. The folder may be Public Reports or My Reports. You can use the button to return to the parent folder. The root folder cannot be used to store resources into.

   The resource table shows the analysis templates in the current directory. Click the column names to change the order of the analysis templates in the table list if required.

2. In the File Name box, enter the name of the analysis template or use the default name, without suffix.

3. Click **OK** to save the analysis template. The saved template is linked to its original catalog.

To save a copy of a Visual Analysis template, click on the toolbar or click > **Save As** to show the Save As dialog, and then do as above.

After saving your analysis template into the server resource tree, you can browse to its directory on the JReport Console > Resources page and run it directly.

To find a newly saved Visual Analysis template version, browse to locate the row that the Visual Analysis template is in on the JReport Console > Resources page, put the mouse pointer over the row and click the **Version** command button on the floating toolbar, the Analysis Versions panel will be displayed showing the versions.

Note: You will not be able to save resources to some locations if you do not have the Write permission.

Integrating JReport Server with a Java Application Server

JReport Server is implemented using Java Servlet technology and Java Server Pages (JSP). These servlets and JSP pages enable you to work with any Java EE compliant application server and administer JReport Server remotely through a web browser.

In order to deploy to an application server, you first have to create a Web Application Archive (WAR) file or an Enterprise Application Archive (EAR) file to include a JReport Server, and then use the application server deployment tools to deploy the WAR/EAR file.

This chapter covers the following:

- Seamless integrated security solution
- Building a WAR/EAR file to include a self-contained JReport Server
- Deploying JReport Server to a Java application server
- Integrating remote JReport Server

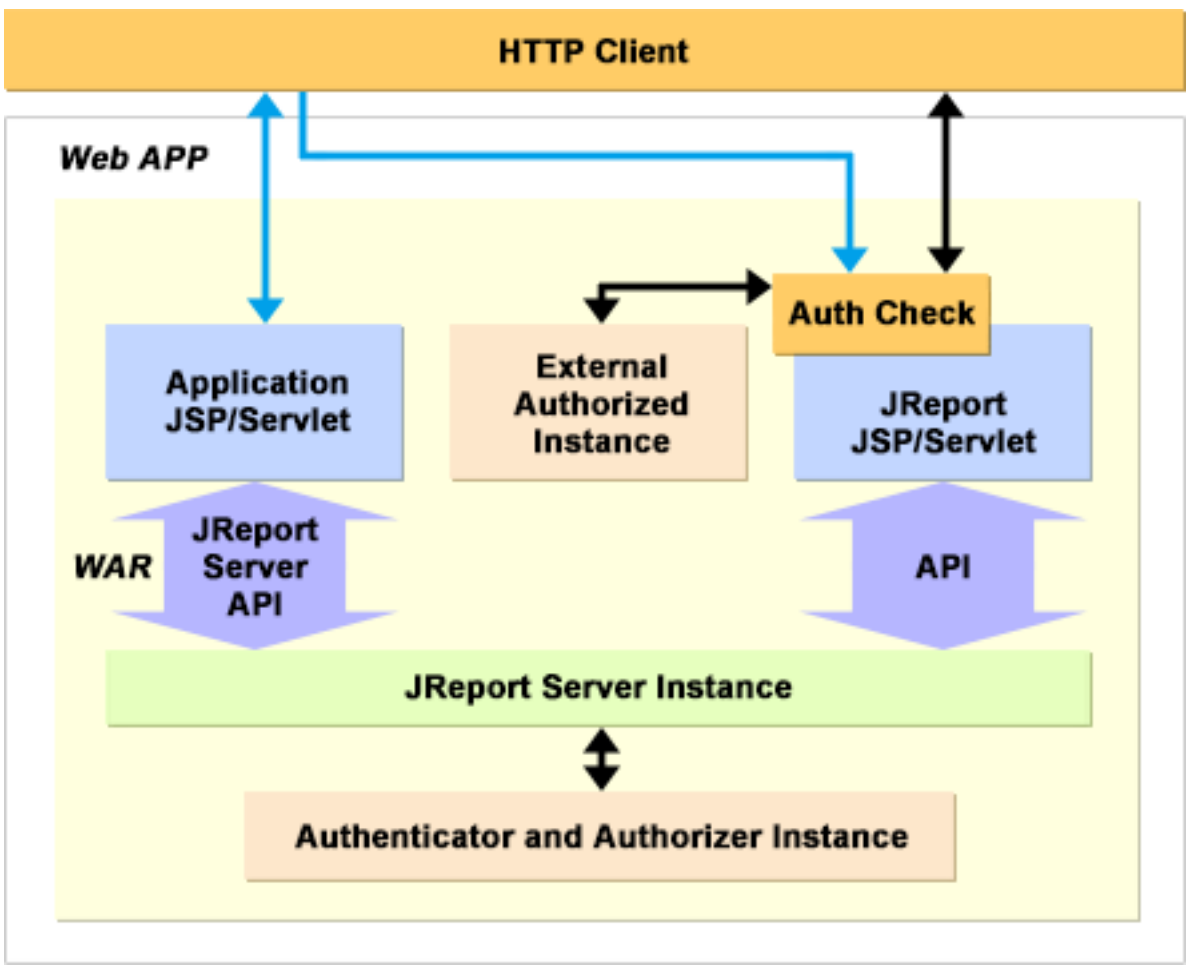
Note: Page Report Studio and Web Report Studio slather dynamic classes, so you probably encounter "OutOfMemoryError: PermGen space" problem when working with them after integration. To solve the problem, you need to add `-XX:MaxPermSize=256m` to JVM or set the number to a bigger one according to your case.

Seamless integrated security solution

As a reporting server, JReport Server protects information via authentication and authorization processes. Furthermore, JReport allows a web application to embed this reporting solution in it seamlessly not only on UI but also with the Java EE technology. In this way, the seamless integrated security solution becomes one of the key solutions of JReport Server.

There are all kinds of scenarios on using JReport solution. However, they can all be categorized into the following two types according to the location of the JReport Server instance.

JReport Server instance is located in the same JVM as the web application



In this scenario, the application includes JReport Server JAR files into the same JVM, and it also includes JReport built-in servlets and JSPs which handle running web and page reports and other reporting services, for example, scheduling reports.

Description of the illustration

In this scenario, the client (HTTP client) most of time will send a request to the portal, JSP or Servlet of the web application, and the web application can either call the public Server API to the server instance directly to run a report and output a report result to file system, or it can re-direct the request to the JReport services provided by the JReport JSPs and Servlets, for example the Page Report Studio JSP and Servlet. JReport JSPs/Servlets will first make sure the request is authenticated and authorized. After which, it will call the internal API method against the JReport Server Instance in the same JVM to fulfill the requirement and return suitable information to the client via JSPs or internally generated output steam.

In the illustration above, you can see that the HTTP client can send a request directly to the application JSP/Servlets or JReport JSPs/Servlets. Before a response is made by the JReport JSP and Servlet, an Auth Check is performed to authenticate the session and then authorize the action. Normally, the built-in authenticator and authorization instance of JReport Server (Instance) is called to perform these checking actions. However, if the application wants to control the process, the web application developer can set up the configuration to ensure that the customized authenticator and authorization instance is used instead.

Pay attention to the RED box: External Authorized instance. This Java class implements JReport jet.server.api.http.HttpExternalAuthorized to provide the authenticated user ID from the session. If this Instance returns a user ID, JReport will pass it to its authenticator to check if it is valid. If the user ID is valid for JReport, JReport will qualify the session of the request, and will not ask for a login again. If this external authorized instance does not return a user ID, JReport will respond the request by asking for a login.

The other RED box - Authenticator and Authorizer instance, can be provided by implementing two other interfaces:

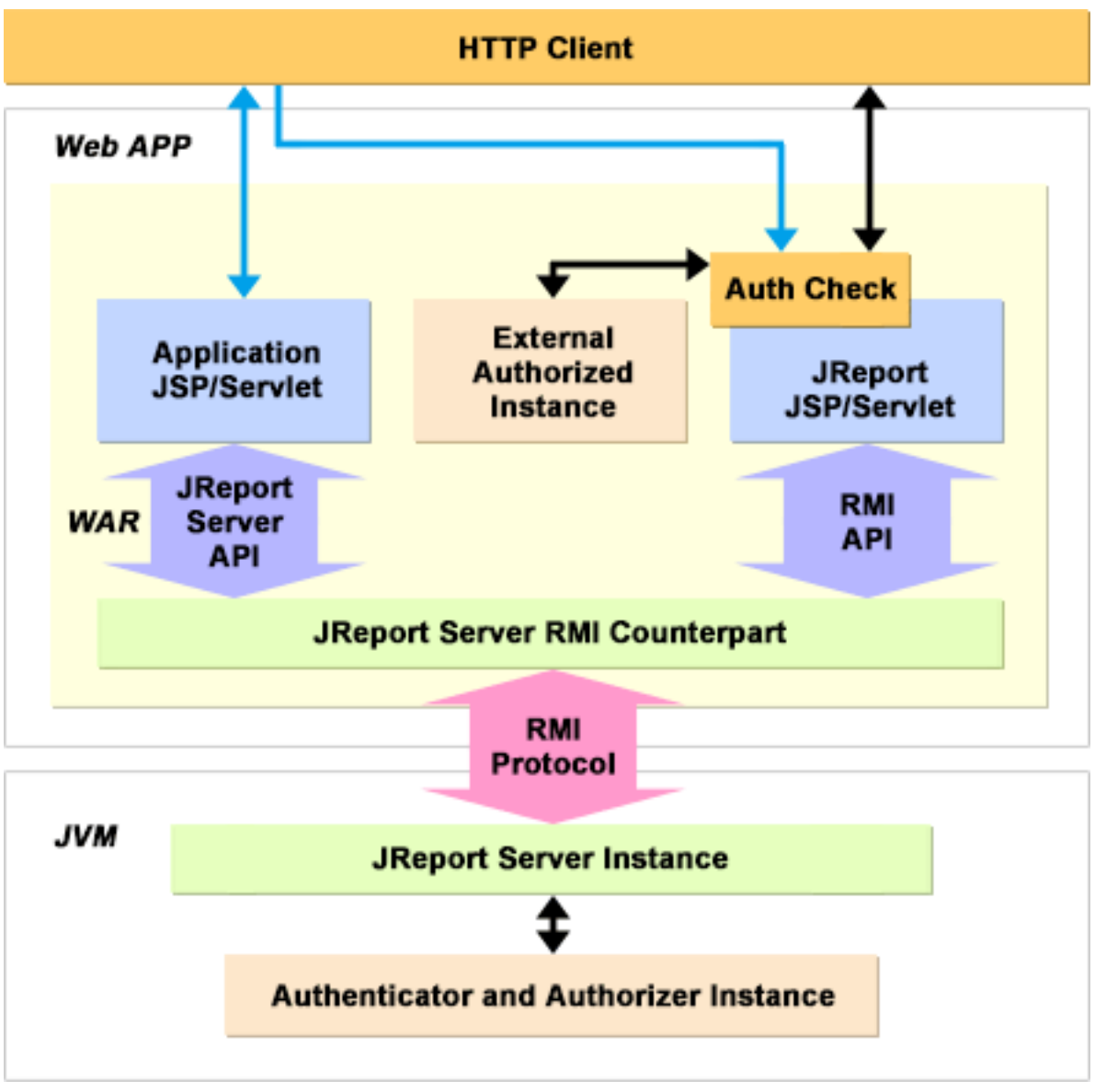
jet.server.api.custom.security.AuthenticationProvider and jet.server.api.custom.security.AuthorizationProvider.

The AuthenticationProvider is used to authenticate the user ID, including whether or not the user ID is valid. The AuthorizationProvider is used to check the privileges of the user against the action that the user requests.

During the auth check process, if the external authorized instance returns a user ID of the session, JReport auth check will continue to send the user ID to the AuthenticatorProvider to check if it is valid or not. If the user is valid, the auth check will qualify the session of the request, and then continue to check if the action is valid for the user by asking the AuthorizationProvider instance.

In general, there is an authentication callback via the implemented interface of External Authorized. Two security check providers can be implemented to seamlessly integrate JReport security into the application.

JReport Server instance is located in a different JVM from the web application



From the web application itself, the architecture is not changed. However, the way that it uses the JReport solution is different since the JReport Server Instance is outside of the Web application server. Inside of the Web Application, the instance is RMI server being called by the web application server or JReport built-in JSP/Servlets for the RMI solution.

Building a WAR/EAR file to include a self-contained JReport Server

There are two ways of creating a WAR or EAR to include a self-contained JReport Server:

- Create a WAR/EAR file using the provided tool makewar.bat/makewar.sh after you have installed a JReport Server. If you are not familiar with JReport Server, it is better to use this way.
- Create a WAR manually. The method is no longer needed but is still available in this release in case you would like to take it.

The self-contained JReport Server is based on a library. The library contains all class packages required by the JReport Server runtime, such as jrenv.jar, JRESServlets.jar, JREngine.jar, and JRWebDesign.jar. In the library, jrenv.jar contains the entire JReport runtime environment, and is the key to the self-contained integration solution. With the self-contained solution, you do not have to specify the JReport Server installation root as the reporthome.

When you create a JReport Server WAR/EAR file using the provided tool, the jrenv.jar package will be automatically put into the WAR/EAR, and will be extracted to the specified reporthome when initializing JReport Server.

The following is the structure of the jrenv.jar package:

jrenv.jar

workspace/ -- This is the root folder.

bin/ -- This folder contains the license file jslc.dat and configuration files, such as LogConfig.properties and redirect.properties.

lib/ -- This folder contains jar files needed by applets, such as view12.jar and chart.jar.

template/ -- This folder contains template files.

profiling/report/ -- This folder contains profiling report files.

jreports/ -- This folder contains demo reports or pre-published reports.

db/ -- This folder contains demo database for demo reports.

help/ -- This folder contains help documents.

The bin/, lib/, and template/ folders are necessary for the JReport runtime, while the profiling/, jreports/, db/ and help/ folders are optional.

Note: There is a parameter in self-contained WAR/EAR - autoDetectServletPath. It is used to dynamically detect and modify servlet path based on context path of self-contained WAR/EAR when deploying the WAR/EAR to a J2EE application server. This property is enabled by default, and the actual servlet path will be concatenating "context path" with "default servlet path" set in server.properties. If you do not want this way, you can disable the feature using either of the following ways:

- Before making your WAR/EAR, set the parameter autoDetectServletPath to false in makewar.xml which is located in `<install_root>\bin`.
- If the WAR/EAR has already been built, go to web.xml, set this parameter autoDetectServletPath to false.

The following are topics covered in the section:

- Building a JReport Server WAR/EAR by tool
- Building a JReport Server WAR manually (deprecated)
- Four ways of integrating JReport Server

Building a JReport Server WAR/EAR by tool

A tool based on the Apache Ant project is provided by JReport Server to build JReport Server WAR/EAR files which contain the full JReport Server runtime environment. The WAR/EAR files can be deployed to any Java EE compliant application server without having to specify a JReport Server installation root as the reporthome.

You can use the default settings to generate a JReport Server WAR/EAR, or you can customize the reporthome and data source for JReport Server before generating the WAR/EAR using the provided tool.

- Creating a JReport Server WAR/EAR
- Specifying reporthome for JReport Server in a Java EE environment
- Specifying a data source for JReport Server in a Java EE environment

Creating a JReport Server WAR/EAR

JReport Server provides a tool for building a WAR or EAR file. The tool is makewar.bat/makewar.sh, and makewar.xml in `<install_root>\bin`.

makewar.xml

This file can be used to specify the following:

- Targets specified to build the WAR/EAR file. They start with the tag `<target name="xxx"...>`. You can modify the target names. By default, the main targets in the makewar.xml are as follows:
  - Making the server runtime environment
  - Making the WAR file for normal or remote integration
  - Making the EAR file
- Temp directories.
  - The temp directory used to save the temp files when building the WAR/EAR. By default, it is `<install_root>\bin\distribute\temp`.
  - The directory which is used to store the generated WAR/EAR file. By default, it is `<install_root>\bin\distribute`.
- The deployment descriptors, such as web.xml and application.xml. The configuration information, such as the database connection information for the WAR/EAR file is stored in these files.

makewar.bat/makewar.sh

The batch/script file used to build a JReport Server WAR/EAR according to the target specified in makewar.xml.

Usage

```
makewar.bat/makewar.sh [Target Name] [-Dpredeploy=ReportFolder] [-Dreporthome=XXX] [-Djrs.remote.host=XXX] [-Djrs.remote.rmiport=XXX] [-Djrs.rmi.auth_file=XXX]
```

Options

- **Target Name**
  The following targets can be performed:
  - buildWar - Specifies to build the JReport Server WAR. It is the default target.
  - buildEar - Specifies to build the JReport Server EAR.
  - buildRemoteWar - Specifies to build the JReport Server WAR for remote integration.
- **-Dpredeploy=ReportFolder**
  Allows you to deploy the reports and catalogs under ReportFolder to the WAR/EAR file.
- **-Dreporthome**
  Specifies the reporthome that will be set into web.xml in the WAR/EAR. If this argument is not set, reporthome will be decided when the WAR/EAR is loaded by the application server and the location will be `%user.home%/.jreport/default`. This argument takes effect only when the target name is buildWar, buildEar, or buildWar4WS.
- **-Djrs.remote.host**
  Specifies the server's RMI host when building a WAR for remote integration. This argument takes effect only when the target name is buildRemoteWar.
- **-Djrs.remote.rmiport**
  Specifies the server's RMI port when building a WAR for remote integration. This argument takes effect only when the target name is buildRemoteWar.
- **-Djrs.rmi.auth_file**
  Specifies the RMI auth file with the absolute file path when building a WAR for remote integration. This argument takes effect only when the target name is buildRemoteWar.

Examples

- Builds the JReport Server WAR file which is defined by makewar.xml (the default target). The generated WAR file is saved to the default directory `<install_root>\bin\distribute`.

  ```
  makewar.bat
  ```

- Builds the JReport Server WAR file, and saves the generated WAR file jreport.war to the default directory `<install_root>\bin\distribute`. When deploying the jreport.war to an application server, the report home will use `C:\JReport`.

  ```
  makewar.bat buildWar -Dreporthome=C:\JReport
  ```

- Builds the JReport Server EAR file, and saves the generated EAR file jreport.ear to the default directory `<install_root>\bin\distribute`.

  ```
  makewar.bat buildEar
  ```

- Builds the JReport Server WAR file, and deploys the reports and catalogs in `C:\myReport` to the WAR file. The generated WAR file jreport.war will be saved in the default directory `<install_root>\bin\distribute`.

  ```
  makewar.bat buildWar -Dpredeploy=c:\myReport
  ```

- Builds the JReport Server WAR file as defined by makewar.xml for remote integration. The generated WAR file is saved to the default directory `<install_root>\bin\distribute`.

  ```
  makewar.bat buildRemoteWar -Djrs.remote.host=127.0.0.1 -Djrs.remote.rmiport=1129 -Djrs.rmi.auth_file=C:\JReport\Server\bin\rmi.auth
  ```

Specifying reporthome for JReport Server in a Java EE environment

JReport Server requires a reporthome as its working space to hold the entire JReport runtime environment, including the server properties, configuration files and resources. The package jrenv.jar that contains the entire JReport runtime environment will be extracted to the specified reporthome when initializing JReport Server. The reporthome can be any location on the disk where JReport Server has read and write privileges.

You do not have to specify a reporthome for JReport Server since it will create a default reporthome location `<user.home>/.jreport/default`.

However, JReport Server enables you to customize the reporthome location before creating the JReport Server WAR/EAR using the provided tool. You can either specify reporthome directly or specify the implementation of the jet.server.api.http.CustomizedServerEnv interface.

Note: Make sure that the reporthome for the integrated JReport Server is different from that of the standalone JReport Server.

Interface CustomizedServerEnv

The jet.server.api.http.CustomizedServerEnv interface can be used for specifying the JReport Server reporthome and for setting the server properties in a Java EE environment. It contains two methods:

- String getReportHome()
- Properties getServerProperties()

If you specify the implementation of this interface, JReport Server will obtain not only reporthome but also server properties.

Three ways of specifying reporthome

There are three ways of specifying reporthome directly or the implementation of the jet.server.api.http.CustomizedServerEnv interface:

- By JVM -D parameter.

  Set the JVM option `-Dreporthome` before starting the application server, for example:

  ```
  -Dreporthome=/home/jreport
  ```

- Invoking the method jet.server.api.http.HttpUtil.initEnv(Properties props).

  For example:

  ```
  Properties props = new Properties();
  props.setProperty("reporthome", "/home/test/JReport");
  HttpUti.initEnv(props);
  ```

- Using ejb-jar.xml/web.xml.

The following introduces several methods of specifying reporthome or implementation of CustomizedServerEnv:

- Using the default reporthome
- Specifying reporthome directly in web.xml or ejb-jar.xml
- Using the default implementations of jet.server.api.http.CustomizedServerEnv
- Using a customized implementation of jet.server.api.http.CustomizedServerEnv

Using the default reporthome

If you do not specify a reporthome, the self-contained JReport Server will create a default working folder. The default working folder is <user.home>/.jreport/default, where <user.home> is the system property user.home retrieved from Java VM. The JReport Server has the Read and Write privileges in this directory. Different OSs have different real paths for <user.home>. For example,

For Windows: `C:\Documents and Settings\username`

For Unix/Linux: `/home/username`

Notes:

- user.home is a system property of the Java VM (`-Duser.home=xxx`). So you can also specify different folders for this JVM option.
- If JReport Server is running on Windows as a service, the username is the user who installed the service or the specified logon user for the service.
- If JReport Server is running as a Unix/Linux Daemon, you can specify the JVM system property `-Duser.home` in the script file that starts JReport Server.

Specifying reporthome directly in web.xml or ejb-jar.xml

It is recommended that you use the <env-entry></env-entry> tags to specify the reporthome directly in the target "web.xml.norpthome" in the makewar.xml file or in ejb-jar.xml. Also, in the target "web.xml.filter", you can specify the reporthome using the <context-param></context-param> tags.

Specifying the reporthome for WAR

You can use one of the two methods listed below to specify the reporthome for WAR:

- In the makewar.xml file, use the <env-entry></env-entry> tags to specify the reporthome in the target "web.xml.norpthome", and then uncommented the setting. For example:

```
<env-entry-name>jreport.rpthome</env-entry-name>
<env-entry-value>/home/jreport</env-entry-value>
<env-entry-type>java.lang.String</env-entry-type>
```

  This is the recommended way to set reporthome since the <env-entry></env-entry> tags are also supported in ejb-jar.xml (if you call the Server API in your EJB).

- In the makewar.xml file, use the <context-param></context-param> tags to specify the reporthome in the target "web.xml.filter", for example:

```
<context-param>
    <param-name>reporthome</param-name>
    <param-value>/home/jreport</param-value>
</context-param>
```

Specifying the reporthome for EAR

The same methods can be used to specify the reporthome of building the EAR file as of building the WAR file. However, because you can wrap WAR and EJB in the EAR file, you should ensure that you put the reporthome information either in the target "web.xml.norpthome" in the makewar.xml file (for the web module) or in ejb-jar.xml (for the EJB module).

In ejb-jar.xml, use the <env-entry></env-entry> tags to specify the reporthome. For example,

```
<env-entry>
    <env-entry-name>jreport.rpthome</env-entry-name>
    <env-entry-value>/home/jreport</env-entry-value>
    <env-entry-type>java.lang.String</env-entry-type>
<env-entry>
```

Using the default implementations of jet.server.api.http.CustomizedServerEnv

The self-contained JReport Server provides two implementations of CustomizedServerEnv. They are jet.server.DefaultServerEnv and jet.server.MultipleInstanceServerEnv.

jet.server.DefaultServerEnv

If you use this implementation, you will not need to specify it in the target "web.xml" in the makewar.xml or in ejb-jar.xml, since it can find the customized reporthome from the `<context-param></context-param>` tags of the target "web.xml" or the `<env-entry></env-entry>` tags of web.xml or ejb-jar.xml.

jet.server.MultipleInstanceServerEnv

This implementation is extended from DefaultServerEnv. It enables finding the reporthome not only from `<context-param></context-param>` or `<env-entry></env-entry>` tags, but also from an external file `<user.home>/.jreportrc`. However, this implementation has three main limitations. They are:

- This implementation must be clearly specified with the `<env-entry></env-entry>` tags in the target "web.xml" in the makewar.xml or in ejb-jar.xml, as follows:

```
<env-entry>
    <env-entry-name>jreport.servenv</env-entry-name>
    <env-entry-value>jet.server.MultipleInstanceServerEnv</env-entry-value>
    <env-entry-type>java.lang.String</env-entry-type>
</env-entry>
```

- The JReport Server must be initialized with JRServerContextListener from a Web module.

  Since in cases of deploying multiple JReport Server instances in one Java EE application server without touching the WAR, such as extracting the WAR, setting reporthome and rebuilding the WAR, JReport Server has to use ServletContext to generate an ID for every instance. JReport Server retrieves javax.servlet.context.tempdir from ServletContext by invoking the getAttribute(String) method, and then uses this value to generate the instance ID.

  For detailed information, see Java Servlet Specification Version 2.3/2.4 SRV.3.7.1 Temporary Working Directories.

- The `<user.home>/.jreportrc` file must be created by JReport Server. You can only edit the file to change the reporthome after the JReport Server initialization.

  Since the instance ID is generated based on a hash code retrieved from javax.servlet.context.tempdir, it cannot be pre-assigned, and is therefore impossible for you to create the `<user.home>/.jreportrc` file. However, once the file has been created, you can edit it to change the reporthome for each instance. The RC file can hold multiple records. The record format should be as follows:

  `jreport.rpthome.<instanceID>=the-instance-report-home`

  The instanceID is created by JReport Server during its first initializing. It is a string of HEX encoded hash value. For example:

  `jreport.rpthome.12345678=/home/user1/.jreport/instance.12345678`

  `jreport.rpthome.12345abc=/home/user1/.jreport/instance.12345abc`

  If JReport Server cannot get the reporthome from CustomizedServerEnv, it will create a default reporthome in `<user.home>/.jreport/default`.

The following is an example of specifying reporthomes when deploying multiple server instances using jet.server.MultipleInstanceServerEnv:

Example: Specifying reporthomes when deploying multiple server instances

JReport Server provides an internally implemented class of the jet.server.api.http.CustomizedServerEnv interface - jet.server.MultipleInstanceServerEnv which supports multiple JReport Server instances in one Java EE application server. The reporthome of each instance can be assigned by the class automatically. To do this, follow the steps below:

1. In the file makewar.xml, use the `<env-entry></env-entry>` tags to specify jet.server.MultipleInstanceServerEnv in the target "web.xml". For example:

```
<env-entry>
    <env-entry-name>jreport.servenv</env-entry-name>
    <env-entry-value>jet.server.MultipleInstanceServerEnv</env-entry-value>
    <env-entry-type>java.lang.String</env-entry-type>
</env-entry>
```

2. Since different JReport instances cannot access the same JReport Server system database at the same time, you will need to create a dbconfig.xml file to store the connection information, and put it in jrenv.jar in `workspace\bin` (you can find the jrenv.jar file after extracting jreport.war) for each JReport Server WAR/EAR.

   For detailed information about modifying the dbconfig.xml, see Configuring the server database.

3. When the JReport Server WAR/EAR has been deployed, JReport will create a file .jreportrc in `<user.home>`, and each JReport instance will read its reporthome from this file. This file must be created by JReport Server, however, you can edit it in order to change the reporthome after it has been created.

Using a customized implementation of jet.server.api.http.CustomizedServerEnv

You can implement the interface and add your class to the generated WAR/EAR file, then use the `<env-entry></env-entry>` tags to specify your implemented class in the target "web.xml" in the makewar.xml file or in ejb-jar.xml.

The following is an example of specifying a customized implementation of CustomizedServerEnv in the target web.xml in the makewar.xml file or in ejb-jar.xml using the `<env-entry></env-entry>` tags. Here the customized implementation of CustomizedServerEnv is named my.JReportServerEnv.

```
<!-- JReport Server calls my.JReportServerEnv to obtain reporthome and server properties.-->
<env-entry>
    <env-entry-name>jreport.servenv</env-entry-name> <!-- must be jreport.servenv-->
    <env-entry-value>my.JReportServerEnv</env-entry-value> <!-- your class name -->
    <env-entry-type>java.lang.String</env-entry-type>
</env-entry>
```

Notes:

- JReport Server will retrieve the reporthome and properties from your implemented class when the WAR/EAR file has been deployed.
- The properties returned from getServerProperties() can be the properties listed in server.properties file in the directory `<install_root>\bin`, JVM System properties, or self-defined ones.

Specifying a data source for JReport Server in a Java EE environment

See Configuring the server database when integrating with an application server for detailed information.

Building a JReport Server WAR manually (deprecated)

This section introduces a method that has been used in earlier versions. If using this method, you will have to specify the JReport Server installation root as the reporthome unless you make the WAR be a self-contained solution.

This section takes creating a WAR file on Unix for example. The instruction is applicable to both Unix and Windows platforms. However, the paths for Windows should use the Windows format, for example, `C:\JReport\Server`, while paths for Unix should use the Unix format, for example, `/opt/JReport/Server`.

It is assumed that JReport Server has been installed to `/opt/JReport/Server`.

Take the following steps to build a JReport Server WAR manually:

1. Create a new directory jreport in the JReport Server installation root: `/opt/JReport/Server/jreport`.
2. Create a sub directory WEB-INF in jreport: `/opt/JReport/Server/jreport/WEB-INF`.
3. Create a web.xml file in the WEB-INF directory as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN" "http://java.sun.com/dtd/
web-app_2_3.dtd">
<web-app>
    <!--By default JReport server detect servlet path-->
    <!--
    <context-param>
    <param-name>autoDetectServletPath</param-name>
    <param-value>false</param-value>
    </context-param>
    -->
    <filter>
        <filter-name>Character Encoding</filter-name>
        <filter-class>jet.server.servlets.CharacterEncodingFilter</filter-class>
        <init-param>
            <param-name>encoding</param-name>
            <param-value>UTF-8</param-value>
        </init-param>
        <init-param>
            <param-name>excludedFiles</param-name>
            <param-value>.css,.js</param-value>
        </init-param>
    </filter>
    <filter>
        <filter-name>JzFileFilter</filter-name>
        <filter-class>com.jinfonet.web.client.RespHeaderFilter</filter-class>
        <init-param>
            <param-name>Content-Type</param-name>
            <param-value>application/x-javascript</param-value>
        </init-param>
        <init-param>
            <param-name>Content-Encoding</param-name>
            <param-value>gzip</param-value>
        </init-param>
    </filter>
    <filter-mapping>
        <filter-name>JzFileFilter</filter-name>
        <url-pattern>*.js</url-pattern>
    </filter-mapping>
    <filter-mapping>
        <filter-name>Character Encoding</filter-name>
        <url-pattern>/*</url-pattern>
    </filter-mapping>
    <listener>
        <listener-class>jet.server.servlets.JRServerContextListener</listener-class>
    </listener>
    <servlet>
        <servlet-name>jrserver</servlet-name>
        <servlet-class>jet.server.servlets.JRServlet</servlet-class>
    </servlet>
    <servlet>
        <servlet-name>sendfile</servlet-name>
        <servlet-class>jet.server.servlets.SendFileServlet</servlet-class>
    </servlet>
    <servlet>
        <servlet-name>dhtml</servlet-name>
        <servlet-class>jet.web.dhtml.DHTMLlet</servlet-class>
    </servlet>
    <servlet>
        <servlet-name>jrdhtml</servlet-name>
        <servlet-class>jet.web.dhtml.DHTMLRunReportlet</servlet-class>
    </servlet>
    <servlet>
        <servlet-name>WebDSServlet</servlet-name>
        <display-name>WebDSServlet</display-name>
        <servlet-class>com.jinfonet.web.client.WebDSServlet</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>jrserver</servlet-name>
        <url-pattern>/jrserver/*</url-pattern>
    </servlet-mapping>
    <servlet-mapping>
        <servlet-name>sendfile</servlet-name>
        <url-pattern>/sendfile/*</url-pattern>
    </servlet-mapping>
    <servlet-mapping>
        <servlet-name>dhtml</servlet-name>
        <url-pattern>/dhtml/*</url-pattern>
    </servlet-mapping>
    <servlet-mapping>
        <servlet-name>jrdhtml</servlet-name>
        <url-pattern>/jrdhtml/*</url-pattern>
    </servlet-mapping>
    <servlet-mapping>
        <servlet-name>WebDSServlet</servlet-name>
        <url-pattern>*.do</url-pattern>
    </servlet-mapping>
    <servlet-mapping>
        <servlet-name>WebDSServlet</servlet-name>
        <url-pattern>*.vt</url-pattern>
    </servlet-mapping>
    <servlet-mapping>
        <servlet-name>WebDSServlet</servlet-name>
        <url-pattern>/vt/*</url-pattern>
    </servlet-mapping>
    <welcome-file-list>
        <welcome-file>index.htm</welcome-file>
        <welcome-file>index.html</welcome-file>
        <welcome-file>index.jsp</welcome-file>
    </welcome-file-list>
    <!-- Define J2EE DataSource resource ref -->
    <!--
    <resource-ref>
        <description>
            Define J2EE DataSource resource for JReport server DB 'systemtables'.
        </description>
        <res-ref-name>jdbc/ds-jreport-systemtables</res-ref-name>
        <res-type>javax.sql.DataSource</res-type>
        <res-auth>Container</res-auth>
    </resource-ref>
    -->
    <!--
    <resource-ref>
        <description>
            Define J2EE DataSource resource for JReport server DB 'realmtables'.
        </description>
        <res-ref-name>jdbc/ds-jreport-realmtables</res-ref-name>
        <res-type>javax.sql.DataSource</res-type>
        <res-auth>Container</res-auth>
    </resource-ref>
    -->
    <!--
    <resource-ref>
        <description>
            Define J2EE DataSource resource for JReport server DB 'profile'.
        </description>
        <res-ref-name>jdbc/ds-jreport-profiling</res-ref-name>
        <res-type>javax.sql.DataSource</res-type>
        <res-auth>Container</res-auth>
    </resource-ref>
    -->
    <login-config>
        <auth-method>BASIC</auth-method>
    </login-config>
    <!-- Notify JReport server to use J2EE DataSource -->
    <!--
    <env-entry>
        <description>
            Notify JReport server DB 'systemtables' to use J2EE DataSource.
        </description>
        <env-entry-name>jreport.datasource.systemtables</env-entry-name>
        <env-entry-value>jndi://jdbc/ds-jreport-systemtables</env-entry-value>
        <env-entry-type>java.lang.String</env-entry-type>
    </env-entry>
    -->
    <!--
    <env-entry>
        <description>
            Notify JReport server DB 'realmtables' to use J2EE DataSource.
        </description>
        <env-entry-name>jreport.datasource.realmtables</env-entry-name>
        <env-entry-value>jndi://jdbc/ds-jreport-realmtables</env-entry-value>
        <env-entry-type>java.lang.String</env-entry-type>
    </env-entry>
    -->
    <!--
    <env-entry>
        <description>
            Notify JReport server DB 'profile' to use J2EE DataSource.
        </description>
        <env-entry-name>jreport.datasource.profiling</env-entry-name>
        <env-entry-value>jndi://jdbc/ds-jreport-profiling</env-entry-value>
        <env-entry-type>java.lang.String</env-entry-type>
    </env-entry>
    -->
    <!-- Specify report home or customized server environment callback -->
    <env-entry>
        <description>
            Specify report home for JReport server directly.
        </description>
        <env-entry-name>jreport.rpthome</env-entry-name>
        <env-entry-value>/home/JettyWarHome</env-entry-value>
        <env-entry-type>java.lang.String</env-entry-type>
    </env-entry>
    <!--
    <env-entry>
        <description>
            Specify customized server environment callback for Jreport server.
        </description>
        <env-entry-name>jreport.servenv</env-entry-name>
        <env-entry-value>jet.server.DefaultServerEnv</env-entry-value>
        <env-entry-type>java.lang.String</env-entry-type>
    </env-entry>
    -->
</web-app>
```

4. Create a directory lib in the `jreport/WEB-INF` directory:

```
mkdir lib
```

5. Create a directory pages in the `jreport/WEB-INF/lib` directory:

```
mkdir lib/pages
```

6. Copy all of the files in `/opt/JReport/Server/lib/pages` to the `jreport/WEB-INF/lib/pages` directory:

```
cp /opt/JReport/Server/lib/pages/* lib/pages
```

7. Create a jar file to include the resources folder which is located in `/opt/JReport/Server` and name it languages.jar. For example, run the following command:

```
jar -cvf languages.jar resources
```

Then put the languages.jar in `/opt/JReport/Server/lib`.

8. Copy the following jar files from `/opt/JReport/Server/lib` to the `jreport/WEB-INF/lib` directory: commons-codec-1.2.jar, jai_codec.jar, jai_core.jar, JREngine.jar, JRESServlets.jar, JRWebDesign.jar, languages.jar, log4j-1.2.8.jar, sac.jar, tar.jar, xercesimpl.jar, xml-apis.jar.

   If you want to export reports to the following formats, you should copy the corresponding jar to the `jreport/WEB-INF/lib` directory:

   - To e-mail or use the e-mail Notification function, copy activation-1.1.jar and mail-1.4.jar.
   - To FTP, copy commons-net-ftp-2.0.0.jar.
   - To PDF and Page Report Result, copy itext_1.5.4.jar.
   - To Excel, copy poiHSSF_151.jar.

9. Copy the index.htm file and the admin, dhtmljsp, images, javascript, jinfonet, skin, and style folders from `/opt/JReport/Server/public_html` to the `/opt/JReport/Server/jreport` directory:

```
cp -r /opt/JReport/Server/public_html/* /opt/JReport/Server/jreport
```

   Notes:

   - The jsp files within the admin folder are used by the JReport Administration pages. Those within the dhtmljsp folder are used when viewing reports in Page Report Studio.
   - If you copy index.htm and these folders mentioned above to a sub folder in `/opt/JReport/Server/jreport`, for example, `/opt/JReport/Server/jreport/sub`, to view reports in Page Report Studio, you need:
     - Modify the server.properties file:

       `web.skin.dir=/jreport/sub/skin`

     - Uncomment the following lines in dhtml.jsp in the dhtmljsp folder and make it adapt to the environment as follows:

       `dhtmlConfig.setDHTMLContextPath(SessionID,"http://localhost:8080/jreport/sub");`
       `dhtmlConfig.setDHTMLJspUrl(SessionID,"/jreport/sub/dhtmljsp/");`
       `dhtmlConfig.setDHTMLServletUrl(SessionID, RptSetId, "/jreport/dhtml");`

     - In the step 10, edit the index.htm file like this:

       `<FRAME name="ind" src="/jreport/sub/jinfonet/index.jsp" frameborder="0">`

   Then go to step 11.

10. Edit the index.htm file and add the context path `/jreport` to the src tag. Note that the path separator character is the Unix style "/" when referencing JSP. The result should be as follows:

```
<FRAME name="ind" src="/jreport/jinfonet/index.jsp" frameborder="0">
```

11. If you are going to create the war for use in Weblogic, you need to check whether there is a weblogic.xml in the `jreport/WEB-INF` directory. If yes, make sure the following line is added in the weblogic.xml:

```
<show-archived-real-path-enabled>true</show-archived-real-path-enabled>
```

   If no, create a weblogic.xml and add it in the `jreport/WEB-INF` directory. The content in the weblogic.xml looks like:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE   weblogic-web-app PUBLIC "-//BEA Systems, Inc.//DTD Web Application 8.1/EN"
"http://www.bea.com/servers/wls810/dtd/weblogic810-web-jar.dtd">

<weblogic-web-app>
    <container-descriptor>
    <show-archived-real-path-enabled>true</show-archived-real-path-enabled>
    </container-descriptor>
</weblogic-web-app>
```

12. Using the following command to create a WAR file named jreport.war:

```
jar -cvf jreport.war index.htm admin dhtmljsp images javascript jinfonet skin style WEB-INF
```

   Note: The jar utility is in the Java home bin directory. If it is not on your path you must call jar with the entire path, for example, `/opt/jdk1.6.0_17/bin/jar.exe`.

To make the WAR be a self-contained solution

To make the WAR include a self-contained JReport Server, except for the above procedure, you need create jrenv.jar and then put it in the `jreport/WEB-INF/lib` directory before creating the WAR.

Use either way to create the jrenv.jar:

- **By the makewar.bat / makewar.sh tool**
  For example: run the command `makewar jrenv.jar`
- **Creating manually**
  Make sure all necessary contents are included and then use a proper tool to package them into a jar file.

See also Building a WAR/EAR file to include a self-contained JReport Server for details about the structure of the jrenv.jar.

Four ways of integrating JReport Server

You can either create a JReport Server WAR/EAR, or create your WAR/EAR and embed a self-contained JReport Server inside it. The following are four ways of building a WAR/EAR file in order to include a self-contained JReport Server:

- Integrating by building a JReport Server WAR
- Integrating by building a JReport Server EAR
- Integrating by building a user WAR and embedding a self-contained JReport Server
- Integrating by building a user EAR and embedding a self-contained JReport Server

Integrating by building a JReport Server WAR

The self-contained JReport Server can be deployed as a single WAR file. After specifying the reporthome and configuring the database information, you can build the WAR file using the tool makewar.bat/makewar.sh in `<install_root>\bin`. To generate the WAR file, execute the command with the buildWar parameter. For example:

```
makewar.bat/makewar.sh buildWar
```

Structure of the JReport Server WAR

For example, here you can create a JReport Server WAR file named jreport.war. The structure of the JReport Server WAR is as follows:

jreport.war

- WEB-INF/
  - web.xml
  - lib/ -- This folder contains all resources in the JReport Server library.
- admin/ -- This folder contains JSP files for the JReport Administration page.
- jinfonet/ -- This folder contains JSP files for the JReport Console page.
- dhtmljsp/ --This folder contains JSP files for viewing DHTML reports.

WEB-INF/web.xml

The following is the default content in the web.xml file of the JReport Server WAR:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems,
Inc.//DTD Web Application 2.3//EN"
"http://java.sun.com/dtd/web-app_2_3.dtd">

<web-app>

    <listener>
        <listener-class>jet.server.servlets.JRServerContextListener</listener-class>
    </listener>

    <servlet>
        <servlet-name>jrserver</servlet-name>
        <servlet-class>jet.server.servlets.JRServlet</servlet-class>
    </servlet>

    <servlet>
        <servlet-name>sendfile</servlet-name>
        <servlet-class>jet.server.servlets.SendFileServlet</servlet-class>
    </servlet>

    <servlet>
        <servlet-name>dhtml</servlet-name>
        <servlet-class>jet.web.dhtml.DHTMLlet</servlet-class>
    </servlet>

    <servlet>
        <servlet-name>help</servlet-name>
        <servlet-class>jet.web.dhtml.JHelplet</servlet-class>
    </servlet>

    <servlet-mapping>
        <servlet-name>jrserver</servlet-name>
        <url-pattern>/jrserver/*</url-pattern>
    </servlet-mapping>

    <servlet-mapping>
        <servlet-name>sendfile</servlet-name>
        <url-pattern>/sendfile/*</url-pattern>
    </servlet-mapping>

    <servlet-mapping>
        <servlet-name>dhtml</servlet-name>
        <url-pattern>/dhtml/*</url-pattern>
    </servlet-mapping>

    <servlet-mapping>
        <servlet-name>help</servlet-name>
        <url-pattern>/help/*</url-pattern>
    </servlet-mapping>

</web-app>
```

Integrating by building a JReport Server EAR

The self-contained JReport Server can be deployed as a single EAR file. After specifying the reporthome and configuring the database information, you can build the EAR file using the tool makewar.bat/ makewar.sh in `<install_root>\bin`. To generate the EAR file, execute the command with the buildEar parameter, for example:

```
makewar.bat/makewar.sh buildEar
```

Structure of the JReport Server EAR

For example, here you can create a JReport Server EAR file named jreport.ear. The structure of the JReport Server EAR is as follows:

jreport.ear

- META-INF/application.xml
- jreport-lib/ -- This folder contains all resources in the JReport Server library.
- jreport.war
  - META-INF/MANIFEST.MF
  - WEB-INF/web.xml
  - admin/
  - jinfonet/
  - dhtmljsp/

META-INF/application.xml

Following the Java EE standard, you should configure the META-INF/application.xml file before deploying the EAR:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE application PUBLIC "-//Sun Microsystems, Inc.//DTD J2EE Application 1.3//EN"
"http://java.sun.com/dtd/application_1_3.dtd">

<application>
    <display-name>Self-contained JReport Server</display-name>
    <module id="JReportWebModule">
        <web>
            <web-uri>jreport.war</web-uri>
            <context-root>jreport</context-root>
        </web>
    </module>
</application>
```

META-INF/MANIFEST.MF of Web module

Since the JReport Server library is included in jreport-lib of the EAR layer, you must specify Class-Path in the META-INF/MANIFEST.MF file. The contents below should be included in the MANIFEST.MF file:

```
Class-Path: jreport-lib/jrenv.jar jreport-lib/JRESServlets.jar jreport-lib/JREngine.
jar ...
```

Class-Path is a list of all packages in the JReport Server library. Each package name should start with the prefix jreport-lib/, and you should use a blank space to separate package names.

WEB-INF/web.xml of Web module

The contents in the web.xml of the JReport Server EAR are the same as the contents in web.xml of the JReport Server WAR. See the section Integrating by building a JReport Server WAR for details.

Integrating by building a user WAR and embedding a self-contained JReport Server

You can embed a self-contained JReport Server into your WAR.

Structure of the user WAR

For example, here you can create a WAR named MyApp.war, and then embed a self-contained JReport Server inside it. The structure of your WAR may be as follows:

MyApp.war

WEB-INF/

web.xml

lib/ -- This folder contains all resources in the JReport Server library and your other jar files.

classes/ -- This folder contains your servlet classes.

../../asset/images/

jsp/

If you put the JReport WAR related JSPs to a sub folder, for example, jreport, and this time the structure of your WAR may be as follows:

MyApp.war

WEB-INF/

web.xml

lib/ -- This folder contains all resources in the JReport Server library and your other jar files.

classes/ -- This folder contains your servlet classes.

../../asset/images/

jreport/

admin/

dhtmljsp/

jinfonet/

...

To run JReport reports, you must do the following configurations:

- Add the following entry into web.xml:

```
<context-param>
<param-name>autoDetectServletPath</param-name>
<param-value>false</param-value>
</context-param>
```

WEB-INF/web.xml in the user WAR

You should add listener into the web.xml as follows:

```
<listener>
<listener-class>jet.server.servlets.JRServerContextListener</listener-class>
</listener>
```

Integrating by building a user EAR and embedding a self-contained JReport Server

You can embed a self-contained JReport Server into your EAR in order to use JReport Server from EJB.

Structure of the user EAR

For example, here you can create an EAR named MyApp.ear and then embed a self-contained JReport Server inside it. In the EAR, there is an EJB module used for initializing JReport Server. The structure of your EAR may be as follows:

MyEAR.ear

- META-INF/application.xml
- jreport-lib/ -- This folder contains all resources in the JReport Server library
- MyEjb.jar
  - META-INF/
    - MANIFEST.MF
    - ejb-jar.xml
  - com/

META-INF/application.xml

Following the Java EE standard, you should configure the META-INF/application.xml file before deploying your EAR:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE application PUBLIC "-//Sun Microsystems, Inc.//DTD J2EE Application 1.3//EN"
"http://java.sun.com/dtd/application_1_3.dtd">

<application>
    <display-name>EJB with Embedded JReport Server</display-name>
    <module id="MyEJBModule">
        <ejb>MyEjb.jar</ejb>
    </module>
</application>
```

META-INF/MANIFEST.MF of EJB module

Since the JReport Server library is included in the jreport-lib folder of the EAR layer, you must specify Class-Path in the META-INF/MANIFEST.MF file. The contents below should be included in the MANIFEST.MF file:

```
Class-Path: jreport-lib/jrenv.jar jreport-lib/JRESServlets.jar jreport-lib/JREngine.
jar ...
```

Class-Path is a list of all packages in the JReport Server library. Each package name should start with the prefix jreport-lib/, and you should use a blank space to separate package names.

META-INF/ejb-jar.xml

If you do not want to set the reporthome for the embedded self-contained JReport Server, but instead want to use JReport's default settings, there is no requirement for configuring the ejb-jar.xml. However, if you want to control the reporthome of the JReport Server, or specify a JNDI data source for JReport Server to use, you should first configure the ejb-jar.xml file using the `<env-entry></env-entry>` tags.

Here is an example:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE ejb-jar PUBLIC "-//Sun Microsystems, Inc.//DTD Enterprise JavaBeans 2.0//EN"
"http://java.sun.com/dtd/ejb-jar_2_0.dtd">

<ejb-jar id="ejb-jar_ID">
    <display-name>MyEJB</display-name>
    <enterprise-beans>
        <session id="JReportEJB">
            <ejb-name>JRptServer</ejb-name>
            <home>demo.JRptServerHome</home>
            <remote>demo.JRptServer</remote>
            <ejb-class>demo.JRptServerBean</ejb-class>
            <session-type>Stateless</session-type>

            <!-- Specify JReport reporthome directly
                <env-entry>
                <env-entry-name>jreport.rpthome</env-entry-name>
                <env-entry-value>/home/jreport</env-entry-value>
                <env-entry-type>java.lang.String</env-entry-type>
                </env-entry>
            -->

            <!-- JReport callback my CustomizedServerEnv -->
            <env-entry>
                <env-entry-name>jreport.servenv</env-entry-name>
                <env-entry-value>demo.JReportServerEnv</env-entry-value>
                <env-entry-type>java.lang.String</env-entry-type>
            </env-entry>

            <transaction-type>Bean</transaction-type>
        </session>
    </enterprise-beans>
</ejb-jar>
```

Deploying JReport Server to a Java application server

After you have created a WAR/EAR file that includes a self-contained JReport Server, you can deploy the WAR/EAR to an application server following the deploying instructions of the application server.

This section provides examples of deploying JReport Server to several leading Java EE application servers. The instructions are applicable to Unix, z/Linux and Windows platforms. However, the paths for Windows should use the Windows format, for example, `C:\JReport\Server`, while paths for Unix and z/Linux should use the Unix and z/Linux format, for example, `/opt/JReport/Server`.

Before going on to the next sections, you should already have reviewed Building a WAR/EAR file to include a self-contained JReport Server.

The following examples are based on the Unix platform with one exception of Sun Application Server on Windows:

- Deploying to IBM WebSphere 8.5.3.3
- Deploying to WebLogic 12c Release 1 (12.1.2)
- Deploying to Tomcat 8.0.23
- Deploying to jboss-as-7.1.1.Final or JBoss EAP 6.4
- Deploying to OC4J 10g R3 (10.1.3.5.0)
- Deploying to Sun Java™ System Application Server Platform Edition 9.1
- Deploying to Jetty 9.3.7
- Deploying to GlassFish V3
- Deploying to Resin 4.0.44
- Deploying to WildFly 9.0.2 or WildFly 10.0.0

Note: It is supported if you change the location of the two folders, skin and dhtmljsp, which are in the `\public_html` directory in the application server side. What is needed is creating a file jrserver.properties in the \WEB-INF directory and then adding the following two properties and providing the correct paths (the context root is excluded):

```
web.skin.dir
web.dhtml_jsp_path
```

Deploying to IBM WebSphere 8.5.3.3

The example directory paths listed below are based on Solaris or Linux (referred to as Unix). The instructions are applicable to both Unix and Windows installations; however, the format of the paths for Windows would use the Windows format, that is, `C:\JReport\Server` instead of `/opt/JReport/Server`.

It is assumed that:

- WebSphere 8.5.3.3 is installed in the `/opt/IBM/WebSphere8.5.3.3/AppServer` directory.
- The JReport Server WAR file jreport.war is located in the `/opt/JReport/Server/bin/distribute` directory. To create the WAR file, refer to the instructions in Building a WAR/EAR file to include a self-contained JReport Server.

To deploy JReport Server to IBM WebSphere:

1. Copy Derby jars in `/opt/JReport/Server/derby` to the `/opt/IBM/WebSphere8.5.3.3/AppServer/lib` directory.
2. Start IBM WebSphere. Use the shell script `/opt/IBM/WebSphere8.5.3.3/AppServer/bin/startServer.sh <servername>` to start the server. The default server name is server1.
3. Access the WebSphere Administrative Console by using the URL: `http://hostname:9060/ibm/console`, where the hostname is host name or IP address, and 9060 is the port number.
4. The login requires user name and password.
5. After successfully log in, expand the **Applications** node, select **Application Types** and then **Websphere enterprise applications**.
6. Click **Install**.
7. Click **Browse** to select the jreport.war file, and then click **Next**.
8. Keep clicking **Next** until you see the requirement for specifying context root.
9. In the Context Root field, type a context path such as **/jreport/**, then click **Next**.
10. Click **Finish** on the Summary page. The installing process may take several minutes, wait until the process is completed.
11. Click **Save**.
12. Select **jreport.war** and then click **Start** to start JReport Server.
13. Access JReport Server using the following URL:

    ```
    http://<hostname>:9080/jreport/jrserver
    http://<hostname>:9080/jreport/admin/index.jsp
    http://<hostname>:9080/jreport/jinfonet/index.jsp
    ```

Troubleshooting

If you run into problems when using JReport Server in IBM WebSphere, send the log files of JReport Server to support@jinfonet.com. The following procedure illustrates how to generate the log files:

1. Type **-Dlogall=true** in the Generic JVM arguments field. Go to **Application servers** > **server1** > **Process Definition** > **Java Virtual Machine** to access this field.
2. Restart the application server, and try to reproduce the problem. After reproducing the problem, send support@jinfonet.com the log files in `reporthome/logs`.

   The WebSphere log files may also help to identify the problem. The most useful one is in `/opt/IBM/WebSphere8.5.3.3/AppServer/profiles/AppSrv01/logs/server1/SystemErr.log`.

Note: For WebSphere Application Server Liberty Profile V8.5.5.2, you need to do the following:

- Configure JNDI so as for the reporthome to be generated successfully. Add the following lines in server.xml located in `${WebSphere_home}/usr/servers/defaultServer`:

  ```
  <featureManager>
  <feature>jndi-1.0</feature>
  </featureManager>
  ```

- Extract jreport.war to `${wlphome}/usr/servers/defaultServer/dropins` and then start WebSphere Application Server Liberty Profile V8.5.5.2. In this way the sample reports will be able to run well.

Deploying to WebLogic 12c Release 1 (12.1.2)

The example directory paths listed below are based on Solaris or Linux (referred to as Unix). The instructions are applicable to both Unix and Windows installations; however, the format of the paths for Windows would use the Windows format, that is, `C:\JReport\Server` instead of `/opt/JReport/Server`.

It is assumed that:

- WebLogic 12c Release 1 (12.1.2) is installed in the `/opt/bea` directory. This is referred to as BEA_HOME in the WebLogic documentation.
- The JReport Server WAR file jreport.war is located in the `/opt/JReport/Server/bin/distribute` directory. To create the WAR file refer to the instructions in Building a WAR/EAR file to include a self-contained JReport Server.

To deploy JReport Server to BEA WebLogic:

1. If you have not already created a WebLogic Domain for JReport Server you must create one before starting the integration.
2. Start WebLogic by running startWeblogic.sh in `/opt/bea/user_projects/domains/domain_name/bin`.
3. Access the WebLogic Administrative Console by using URL `http://hostname:7001/console/`, where the hostname is host name or IP address, and 7001 is the port number.
4. After your successful login, in the Domain Structure panel on the left, click **Deployments** node.
5. In the Summary of Deployments panel, click **Install**.
6. In the Install Application Assistant panel, click the **upload your file(s)** link.
7. In the Deployment Archive section, click **Browse** to select the **jreport.war** file in `C:\JReport\Server\bin\distribute`, and then click **Next**.
8. Keep clicking **Next** until the Finish button is enabled, and then click **Finish**.
9. Start JReport Server and then access it using the following URL:

   `http://localhost:7001/jreport/`

Troubleshooting

If you run into problems when using JReport Server in BEA WebLogic, send the log files of JReport Server to support@jinfonet.com. The following procedure illustrates how to generate the log files:

1. Add **`-Dlogall=true`** on the same line as -Dreporthome in the `startWebLogic.sh` shell script.
2. Restart the application server, and then try to reproduce the problem. After reproducing the problem, send support@jinfonet.com the log files in `reporthome/logs`.

   The WebLogic log file may also help to identify the problem. It is `/opt/bea/user_projects/domains/domain_name/logs`.

Deploying to Tomcat 8.0.23

The example directory paths listed below are based on Solaris or Linux (referred to as Unix). The instructions are applicable to both Unix and Windows installations; however, the format of the paths for Windows would use the Windows format, that is, `C:\JReport\Server` instead of `/opt/JReport/Server`.

It is assumed that:

- Tomcat 8.0.23 is installed in the `/opt/apache-tomcat-8.0.23` directory.
- The JReport Server WAR file jreport.war is located in the `/opt/JReport/Server/bin/distribute` directory. To create the WAR file refer to the instructions in Building a WAR/EAR file to include a self-contained JReport Server.

To deploy JReport Server to Tomcat 8.0.23:

1. Ensure that Tomcat is shut down.
2. Copy the Web Application Archive jreport.war to `/opt/apache-tomcat-8.0.23/webapps`.
3. Start Tomcat by running the `startup.sh` script.
4. Access JReport Server using the following URLs:

   ```
   http://hostname:8080/jreport/jrserver
   http://hostname:8080/jreport/admin/index.jsp
   http://hostname:8080/jreport/jinfonet/index.jsp
   ```

Troubleshooting

If you run into problems when using JReport Server in Tomcat, send the log files of JReport Server to support@jinfonet.com. The following procedure illustrates how to generate the log files:

1. Modify the catalina.sh file in `/opt/apache-tomcat-8.0.23/bin`.

   In the file catalina.sh, add **`-Dlogall=true`** after the reporthome definition:

   ```
   JAVA_OPTS="-Dreporthome=/opt/JReport/Server -Dlogall=true"
   Cygwin=false
   ```

   Or if no reporthome is specified, add as follows:

   ```
   JAVA_OPTS=-Dlogall=true
   Cygwin=false
   ```

2. After editing catalina.sh, start Tomcat.
3. To get the information about the JReport Server environment, you can access `http://hostname:8080/jreport/admin/info.jsp?cmd=info`.
4. Save the output to a file.
5. After reproducing the problem, send support@jinfonet.com the log files in `reporthome/logs`.

   The Tomcat log files may also help to identify the problem. The most useful one is `/opt/apache-tomcat-8.0.23/logs/catalina.out`.

Deploying to jboss-as-7.1.1.Final or JBoss EAP 6.4

The example directory paths listed below are based on Solaris or Linux (referred to as Unix). The instructions are applicable to both Unix and Windows installations; however, the format of the paths for Windows would use the Windows format, that is, `C:\JReport\Server` instead of `/opt/JReport/Server`.

The following takes jboss-as-7.1.1.Final as an example.

It is assumed that:

- jboss-as-7.1.1.Final is installed in the `/opt/jboss-as-7.1.1.Final` directory.
- The JReport Server WAR file jreport.war is located in the `/opt/JReport/Server/bin/distribute` directory. To create the WAR file refer to the instructions in Building a WAR/EAR file to include a self-contained JReport Server.

To deploy JReport Server to jboss-as-7.1.1.Final:

1. Ensure that JBoss is shut down by running the script `/opt/jboss-as-7.1.1.Final/bin/shutdown.sh -S`.
2. To resolve the issue that JBoss AS 7/EAP 6 cannot locate jrenv.jar, when building a WAR/EAR to deploy to JBoss AS 7/EAP 6, add *-Djbossas7=true* in makewar.bat/sh.
3. After jreport.war is generated, you need to create a file named *iboss-deployment-structure.xml* in the jreport.war/META-INF directory. The file should contain the following contents:

```
<?xml version="1.0" encoding="UTF-8"?>

          <jboss-deployment-structure>

                <deployment>
                <exclusions>
                <module name="org.apache.log4j" />
                </exclusions>


                <dependencies>
                <system>
                <paths>
                <path name="com/sun/org/apache/xml/internal/security/
                utils"/>
                </paths>
                </system>
                </dependencies>
                </deployment>

          </jboss-deployment-structure>
```

4. Deploying jreport.war to JBoss AS 7 costs some time. When there is a setting *deployment-timeout="60"* in `<JBoss_home>/standalone/configuration/standalone.xml`, change the value to 180 (seconds) or a bigger number.
5. Start JBoss by running the standalone.sh script.
6. Access JReport Server using the following URLs:

   ```
   http://localhost:8080/jreport/jrserver
   http://localhost:8080/jreport/admin/index.jsp
   http://localhost:8080/jreport/jinfonet/index.jsp
   ```

7. If formulas cannot be compiled when running sample reports, you need to use JReport Designer to compile formulas and then publish the reports to the integrated JReport Server once again.

Troubleshooting

If you run into problems when using JReport Server in JBoss, send the log files of JReport Server to support@jinfonet.com. The following procedure illustrates how to generate the log files:

1. Modify the file run.sh in `/opt/jboss-as-7.1.1.Final/bin`.

   In the file run.sh, add **`-Dlogall=true`** after the reporthome definition:

   ```
   "$JAVA" $JAVA_OPTS \
   -classpath "$JBOSS_CLASSPATH" -Dreporthome=/opt/JReport/Server \
   -Dlogall=true  \
   org.jboss.Main "$@"
   ```

2. After editing run.sh, start JBoss using the modified file.
3. After reproducing the problem, send support@jinfonet.com the log files in `reporthome/logs`.

   The JBoss log files may also help to identify the problem. The most useful one is `/opt/jboss-as-7.1.1.Final/server/default/log/server.log`.

Deploying to OC4J 10g R3 (10.1.3.5.0)

It is assumed that the JReport Server WAR file jreport.war is located in the `/opt/JReport/Server/bin/distribute` directory. To create the WAR file refer to the instructions in Building a WAR/EAR file to include a self-contained JReport Server.

To deploy JReport Server to OC4J 10g R3 (10.1.3.5.0):

1. Start OC4J.
2. On the **Applications** tab, click **Deploy**.
3. In the Archive Location section, click Browse to select the **jreport.war** file in `/opt/JReport/Server/bin/distribute`. Then click **Next**.
4. Type **JRServer** in the Application Name field, and type a context path such as **/jreport** in the Context Root field. Then click **Next**.
5. Click **Deploy** to deploy the WAR.
6. Access JReport Server using the following URLs:

   ```
   http://localhost:port/jreport/jrserver
   http://localhost:port/jreport/admin/index.jsp
   http://localhost:port/jreport/jinfonet/index.jsp
   ```

Troubleshooting

If you run into problems when using JReport Server in the Oracle oc4j Application Server, send the log files of JReport Server to support@jinfonet.com. The following procedure illustrates how to generate the log files:

1. Type **-Dlogall=true** in oc4j located in `/oc4jhome/bin`.
2. Restart OC4J, and then try to reproduce the problem.
3. After reproducing the problem, send support@jinfonet.com the log files in `reporthome/logs`.

Deploying to Sun Java™ System Application Server Platform Edition 9.1

The example directory paths listed below are based on Windows. The instructions are applicable to both Unix and Windows installations; however, the format of the paths for Unix would use the Unix format, that is, `/opt/JReport/Server` instead of `C:\JReport\Server`.

It is assumed that:

- Sun Java™ System Application Server Platform Edition 9.1 is installed in the `C:\Sun\AppServer` directory.
- It is assumed that the JReport Server WAR file jreport.war is located in the `C:\JReport\Server\bin\distribute` directory. To create the WAR file refer to the instructions in Building a WAR/EAR file to include a self-contained JReport Server.

To deploy JReport Server to Sun Java™ System Application Server Platform Edition 9.1:

1. Update Sun Application Server's Derby jars by using the lib folder in `C:\JReport\Server\derby` to replace the lib folder in `C:\Sun\AppServer\javadb\lib`.
2. Start the Sun Application Server by selecting **Start** > **Programs** > **Sun Microsystems** > **Application Server PE** > **Start Default Server.**
3. Launch the Admin Console by selecting **Start** > **Programs** > **Sun Microsystems** > **Application Server PE** > **Admin Console.**
4. In the left console tree, expand the **Applications** node, then click **Web Applications**.
5. In the **Web Applications** page, click **Deploy**.
6. Select the radio button before **Local packaged file or directory that is accessible from the Application Server**, then click **Browse Files** to select the WAR file **jreport.war**.
7. Use the default settings and click **OK**. You will find a new application jreport is listed.
8. Access JReport Server using the following URLs:

   ```
   http://<hostname>:8080/jreport/jrserver
   http://<hostname>:8080/jreport/jinfonet/index.jsp
   http://<hostname>:8080/jreport/admin/index.jsp
   ```

Troubleshooting

If you run into some problems when using the Sun Application Server, send the log files of JReport Server to support@jinfonet.com. The following procedure illustrates how to generate the log files:

1. Start the Sun Application Server, and then launch the Admin Console.
2. In the console tree, click **Application Server**.
3. Go to the JVM Settings tab, and then click **JVM Options**.
4. In the JVM Option field, click **Add JVM Option**, and then type **-Dlogall=true**.
5. Click **Save** to save your changes.
6. Restart Sun Application Server and try to reproduce the problem.
7. After reproducing the problem, send support@jinfonet.com the log files in `reporthome/logs`.

Deploying to Jetty 9.3.7

The example directory paths listed below are based on Solaris or Linux (referred to as Unix). The instructions are applicable to both Unix and Windows installations; however, the format of the paths for Windows would use the Windows format, that is, `C:\JReport\Server` instead of `/opt/JReport/Server`.

It is assumed that the JReport Server WAR file jreport.war is located in the `/opt/JReport/Server/bin/distribute` directory. To create the WAR file refer to the instructions in Building a WAR/EAR file to include a self-contained JReport Server.

To deploy JReport Server to Jetty 9.3.7:

1. If you are integrating JReport Server with Jetty in a non-remote environment, you need to remove or comment the following codes in the web.xml of jreport.war:

   ```
   <login-config>
   <auth-method>BASIC</auth-method>
   </login-config>
   ```

2. Add jreport.war to the `/opt/Jetty9.3.7/webapps` directory.

3. Start Jetty 9.3.7.

4. Access JReport Server using the following URLs:

   ```
   http://<hostname>:8080/jreport/jrserver
   http://<hostname>:8080/jreport/admin/index.jsp
   http://<hostname>:8080/jreport/jinfonet/index.jsp
   ```

Deploying to GlassFish V3

The example directory paths listed below are based on Solaris or Linux (referred to as Unix). The instructions are applicable to both Unix and Windows installations; however, the format of the paths for Windows would use the Windows format, that is, `C:\JReport\Server` instead of `/opt/JReport/Server`.

It is assumed that the JReport Server WAR file jreport.war is located in the `/opt/JReport/Server/bin/distribute` directory. To create the WAR file refer to the instructions in Building a WAR/EAR file to include a self-contained JReport Server.

To deploy JReport Server to GlassFish V3:

1. Start the GlassFish in the default domain1 and then launch the Admin Console**.**
2. Click the **Applications** node on the left.
3. Click **Deploy** on the displayed page.
4. Click **Choose File** to select the WAR file **jreport.war**.
5. Leave Application Name and Context Root as jreport and jreport. Then click **OK**.
6. Expand the Deployment node on the left and you will see a new node named jreport. Click **jreport** and then on the displayed page click **Save**.
7. In the console tree, click **Configuration**. Go to the JVM Settings tab, and then click **JVM Options**. In the JVM Options section, click **Add JVM Option**, and then type **-Djava.awt.headless=true**. Click **Save** to save your changes.

   You need not add this JVM option if you are using Windows.
8. Restart GlassFish, and then start the application **jreport**.
9. Access JReport Server using the following URLs:

   ```
   http://<hostname>:8080/jreport
   http://<hostname>:8080/jreport/admin/index.jsp
   http://<hostname>:8080/jreport/jinfonet/index.jsp
   ```

Note: GlassFish 3.1.2 uses a different Derby driver from JReport which leads to that the JReport War file does not work in GlassFish 3.1.2. To solve this, you can replace the files in `/opt/glassfish3.1.2/javadb/lib` with the jar files in `/opt/JReport/Server/derby/lib`, or use the folder `/opt/glassfish3.0.2/javadb/lib` to replace the folder `/opt/glassfish3.1.2/javadb/lib` if GlassFish 3.0.2 resources are available.

Troubleshooting

If you run into some problems when using the GlassFish V3, send the log files of JReport Server to support@jinfonet.com. The following procedure illustrates how to generate the log files:

1. Start the GlassFish in the default domain1 and then launch the Admin Console.
2. In the console tree, click **Configuration**.
3. Go to the JVM Settings tab, and then click **JVM Options**.
4. In the JVM Options section, click **Add JVM Option**, and then type **-Dlogall=true**. Click **Save** to save your changes.
5. Restart GlassFish and try to reproduce the problem.
6. After reproducing the problem, send support@jinfonet.com the log files in `reporthome/logs`.

   The GlassFish log file may also help to identify the problem. It is /opt/glassfish/domains/domain1/logs/server.log.

Deploying to Resin 4.0.44

The example directory paths listed below are based on Solaris or Linux (referred to as Unix). The instructions are applicable to both Unix and Windows installations; however, the format of the paths for Windows would use the Windows format, that is, `C:\JReport\Server` instead of `/opt/JReport/Server`.

It is assumed that:

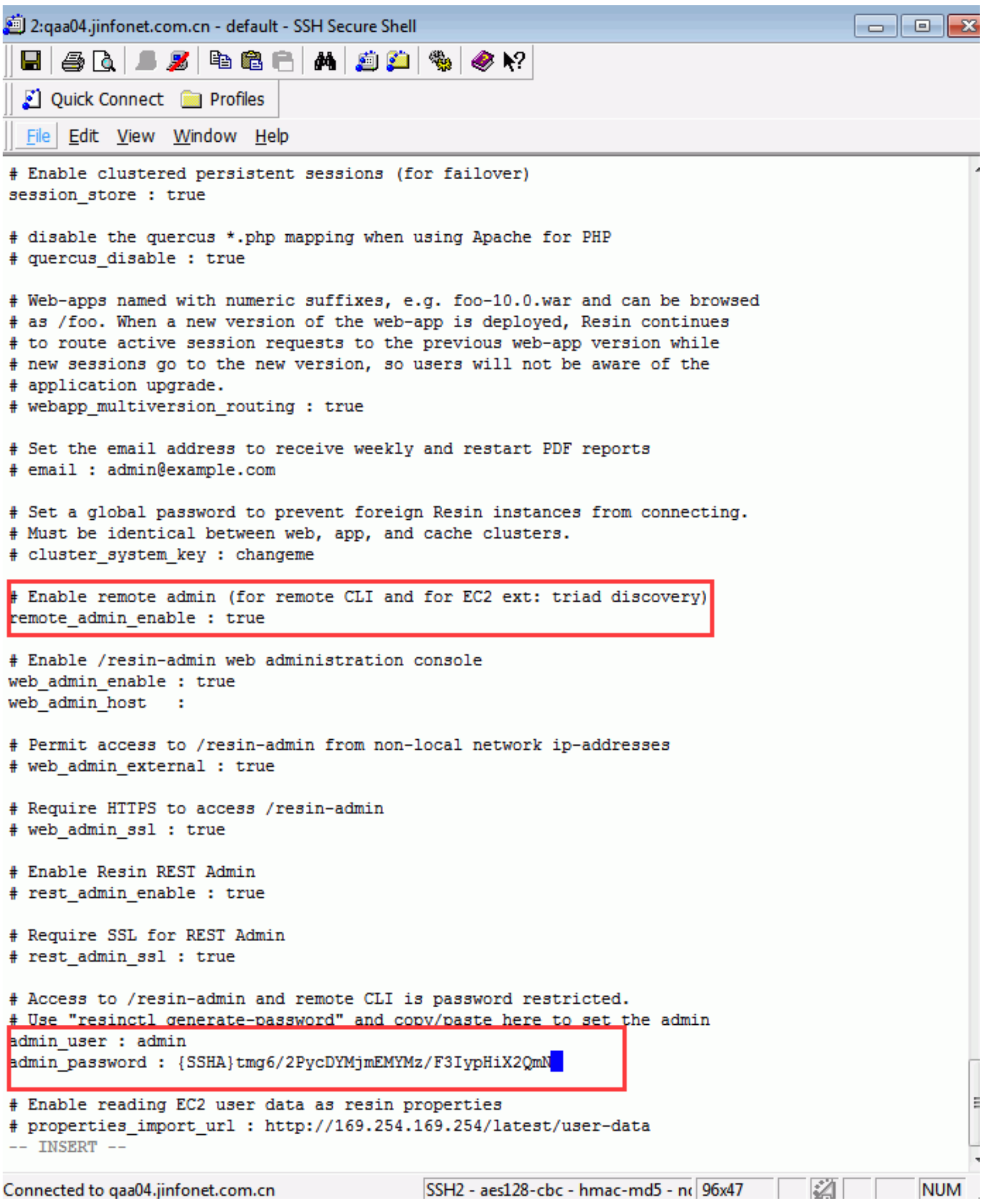
- Resin 4.0.44 is installed in the `/opt/resin` directory.
- The JReport Server WAR file jreport.war is located in the `/opt/JReport/Server/bin/distribute` directory. To create the WAR file refer to the instructions in Building a WAR/EAR file to include a self-contained JReport Server.

To deploy JReport Server to Resin 4.0.44:

1. Generate the password.

```
[root@qaa04 bin]# ./resin.sh password-generate --user admin
Enter password:
Verify password:
admin_user : admin
admin_password : {SSHA}tmg6/2PycDYMjmEMYMz/F3IypHiX2QmN
[root@qaa04 bin]#
```

2. Modify resin.properties in `/opt/resin/conf`. Set remote_admin_enable to true and add the generated password.



3. Start Resin 4.0.44 by running the resin.sh script in `/opt/resin/bin`.
4. Log onto the Resin 4.0.44 Administration Console with the specified user name and password using the URL `http://localhost:8080/resin-admin/index.php`.
5. In the deploy tab, click **Browse** to select the file jreport.war. In the Name text box, type in **jreport**. Then click **Deploy**.
6. Access JReport Server using the following URLs:

```
http://localhost:8080/jreport/jrserver
http://localhost:8080/jreport/admin/index.jsp
http://localhost:8080/jreport/jinfonet/index.jsp
```

Troubleshooting

If you run into problems when using JReport Server in Resin, send the log files of JReport Server to support@jinfonet.com. The following procedure illustrates how to generate the log files:

1. Modify the file resin.sh in `/opt/resin/bin` by adding **`-Dlogall=true`** after the reporthome definition:

```
"$JAVA" $JAVA_OPTS \
-classpath "$RESIN_CLASSPATH" -Dreporthome=/opt/JReport/Server \
-Dlogall=true  \
org.resin.Main "$@"
```

2. Run the modified resin.sh to start Resin.
3. After reproducing the problem, send support@jinfonet.com the log files in `reporthome/logs`.

   The Resin log files may also help to identify the problem. The most useful one is `/opt/Resin/server/default/log/server.log`.

Deploying to WildFly 9.0.2 or WildFly 10.0.0

The example directory paths listed below are based on Solaris or Linux (referred to as Unix). The instructions are applicable to both Unix and Windows installations; however, the format of the paths for Windows would use the Windows format, that is, `C:\JReport\Server` instead of `/opt/JReport/Server`.

The following takes WildFly 9.0.2 as an example.

It is assumed that:

- WildFly 9.0.2 is installed in the `/opt/wildfly` directory.
- The JReport Server WAR file jreport.war is located in the `/opt/JReport/Server/bin/distribute` directory. To create the WAR file refer to the instructions in Building a WAR/EAR file to include a self-contained JReport Server.

To deploy JReport Server to WildFly 9.0.2:

1. Start WildFly 9.0.2 by running the standalone.sh script in `/opt/wildfly/bin`.
2. By default WildFly 9.0.2 is now distributed with security enabled for the management interfaces, this means that before you connect using the administration console you need to add a new user, this can be achieved by using the `add-user.sh` script in the bin folder. You will then be prompted to set the user name and password for the new user.
3. Log onto the WildFly 9.0.2 Administration Console with the specified user name and password using the URL `http://localhost:9990/console`.
4. In the Deployments tab, click **Add**. In the Create Deployment dialog, browse to select the file jreport.war. Then click **Next**.
5. In the step 2 of the Create Deployment dialog, check the **Enable** option and then click **Save**.
6. Upon the successful deployment of the JReport Server WAR file jreport.war, access JReport Server using the following URLs:

   ```
   http://localhost:8080/jreport/jrserver
   http://localhost:8080/jreport/admin/index.jsp
   http://localhost:8080/jreport/jinfonet/index.jsp
   ```

Troubleshooting

If you run into problems when using JReport Server in WildFly, send the log files of JReport Server to support@jinfonet.com. The following procedure illustrates how to generate the log files:

1. Modify the file standalone.sh in `/opt/wildfly/bin`.

   In the file standalone.sh, add **`-Dlogall=true`** after the reporthome definition:

   ```
   "$JAVA" $JAVA_OPTS \
   -classpath "$WILDFLY_CLASSPATH" -Dreporthome=/opt/JReport/Server \
   -Dlogall=true  \
   org.wildfly.Main "$@"
   ```

2. After editing standalone.sh, start WildFly using the modified file.
3. After reproducing the problem, send support@jinfonet.com the log files in `reporthome/logs`.

   The WildFly log files may also help to identify the problem. The most useful one is `/opt/wildfly/server/default/log/server.log`.

Notes:

- To resolve the issue that WildFly 9.0.2 cannot locate jrenv.jar, when building a WAR/EAR to deploy to WildFly 9.0.2, add *-Djbossas7=true* in makewar.bat/sh.
- After jreport.war is generated, you need to create a file named *iboss-deployment-structure.xml* in the jreport.war/META-INF directory. The file should contain the following contents:

  ```
  <?xml version="1.0" encoding="UTF-8"?>
  <jboss-deployment-structure>
  <deployment>
  <exclusions>
  <module name="org.apache.log4j" />
  </exclusions>
  </deployment>
  </jboss-deployment-structure>
  ```

Integrating remote JReport Server

Normally, JReport servlets are only integrated with other applications on the same machine. However, you can now implement JReport Remote Server API in your JSPs, and integrate the JSPs with the application server to call JReport Server, which is running on a different machine.

Notes:

- In a remote integration environment, the options for publishing resources are hidden since they are not supported by JReport JSPs. If you want to publish reports or catalogs to JReport Server, use one of the following ways:
  - Access the JReport Server (not the remote server) Administration page with 8889 as the default port to perform publish work.
  - Copy the report or catalog files to the computer where JReport Server (not the remote server) is located, and then call the RMI API to publish them.
  - Publish the report or catalog files from JReport Designer to the JReport Server.
- In a remote integration environment, running reports to the Applet format is not supported.
- In a remote integration environment, the two tabs Monitor and Data are hidden on JReport Administration page since they are not supported.
- You can change the location of the two folders, skin and dhtmljsp, in the \public_html directory in the application server side. What is required is to create a file jrserver.properties in the \WEB-INF directory and then add the following two properties and provide the correct paths (the context root is excluded):

  ```
  web.skin.dir
  web.dhtml_jsp_path
  ```

See the following cases:

- Integrating remote JReport Server with IBM WebSphere 8.5.3.3 by a WAR file
- Integrating remote JReport Server with WebLogic 11g Release 1 (10.3.2) by a WAR file

Related Topics:

- Overall Remote Server APIs & unified JSPs

Integrating remote JReport Server with IBM WebSphere 8.5.3.3 by a WAR file

Here is an example illustrating the case of using JSPs based on Remote Server APIs to integrate with IBM WebSphere 8.5.3.3.

It is assumed that:

- WebSphere 8.5.3.3 is installed in `C:\WebSphere` in computer A.
- JReport Server is installed in `C:\JReport\Server` in computer B. The computer IP is 127.0.0.1.

Take the following steps to integrate remote JReport Server with IBM WebSphere:

1. Generate a WAR file.
2. Configure JReport Server.
3. Deploy the WAR file.

Below show the details for each step:

Generating the WAR file

1. In computer B, use the tool makewar.bat to build the JReport Server WAR file as defined by makewar.xml for remote integration. Both makewar.bat and makewar.xml are located in `C:\JReport\Server\bin`. Run the following commands in DOS window and the generated WAR file remote.war will be saved to the directory `C:\JReport\Server\bin\distribute`.

   ```
   makewar.bat buildRemoteWar -Djrs.remote.host=127.0.0.1 -Djrs.remote.rmiport=1129 -Djrs.rmi.auth_file=C:\JReport\Server\bin\rmi.auth
   ```

2. Copy the rmi.auth file from `C:\JReport\Server\bin` in computer B to `C:\JReport\Server\bin` in computer A.

Configuring JReport Server

1. Make sure JReport Server has been started once in order that the server.properties file is generated.
2. Change server.properties file in `C:\JReport\Server\bin` as follows:

   ```
   server.rmiserver.enable=true
   server.rmiadminservice.enable=true
   ```

Deploying the WAR file

1. In computer A, start IBM WebSphere.
2. In computer B, access the WebSphere Administrative Console by using the URL: `http://hostname:9060/ibm/console`, where the hostname is computer A's host name or IP address, and 9060 is the port number.
3. After successfully log in, expand the **Applications** node, select **Application Types** and then **Websphere enterprise applications**.
4. Click **Install**.
5. Click **Browse** to select the remote.war file, and then click **Next**.
6. Keep clicking **Next** until you see the requirement for specifying context root.
7. In the Context Root field, type a context path such as **/remote/**, then click **Next**.
8. Click **Finish** on the Summary page. The installing process may take several minutes, wait until the process is completed.
9. Click **Save**.
10. Select **remote.war** and then click **Start**.
11. Access JReport Server using the following URL:

    `http://hostname:9080/remote/jinfonet/default.jsp`

    Here the hostname is computer A's host name or IP address.

Integrating remote JReport Server with WebLogic 11g Release 1 (10.3.2) by a WAR file

Here is an example illustrating the case of using JSPs based on Remote Server APIs to integrate with WebLogic 11g Release 1 (10.3.2).

It is assumed that:

- WebLogic is installed in `C:\bea` in computer A.
- JReport Server is installed in `C:\JReport\Server` in computer B. The computer IP is 127.0.0.1.

Take the following steps to integrate remote JReport Server with WebLogic:

1. Generate a WAR file.
2. Configure JReport Server.
3. Deploy the WAR file.

Below show the details of each step:

Generating a WAR file

1. In computer B, use the tool makewar.bat to build the JReport Server WAR file as defined by makewar.xml for remote integration. Both makewar.bat and makewar.xml are located in `C:\JReport\Server\bin`. Run the following commands in DOS window and the generated WAR file remote.war will be saved to the directory `C:\JReport\Server\bin\distribute`.

   ```
   makewar.bat buildRemoteWar -Djrs.remote.host=127.0.0.1 -Djrs.remote.rmiport=1129 -Djrs.rmi.auth_file=C:\JReport\Server\bin\rmi.auth
   ```

2. Copy the rmi.auth file from `C:\JReport\Server\bin` in computer B to `C:\JReport\Server\bin` in computer A.

Configuring JReport Server

1. Make sure JReport Server has been started once in order that the server.properties file is generated.
2. Change server.properties file in `C:\JReport\Server\bin` as follows:

   ```
   server.rmiserver.enable=true
   server.rmiadminservice.enable=true
   ```

Deploying the WAR file

1. If you have not already created a WebLogic Domain for JReport Server you must create one before starting the integration.
2. In computer A, start WebLogic by running startWeblogic.sh in `C:\bea\user_projects\domains\domain_name\bin`.
3. In computer B, access the WebLogic Administrative Console by using URL `http://hostname:7001/console/`, where the hostname is computer A's host name or IP address, and 7001 is the port number.
4. After your successful login, in the Domain Structure panel on the left, click **Deployments** node.
5. In the Summary of Deployments panel, click **Install**.
6. In the Install Application Assistant panel, click the **upload your file(s)** link.
7. In the Deployment Archive section, click **Browse** to select the **remote.war** file in `C:\JReport\Server\bin\distribute`, and then click **Next**.
8. Keep clicking **Next** until the Finish button is enabled, and then click **Finish**.
9. Start JReport Server in computer B. Then go to computer A and access JReport Server using the following URL:

   ```
   http://localhost:7001/remote/
   ```

JReport Server Cluster

A JReport Server Cluster is a distributed cluster in which a group of servers work together to provide cluster-wide shared resources, security, schedules and version services that manage multiple versions of resources. In a JReport Server Cluster, all clustered servers play exactly the same role. New servers can be added to the existing cluster or removed from the cluster at any time. Dynamic load balancing ensures that all servers in the cluster are utilized efficiently and JReport Server Monitor allows administrators to manage utilization for the entire cluster and add and remove resources as needed. Notifications can be set up to ensure administrators are contacted if any servers encounter problems.

This chapter shows you the infrastructure of the JReport Server Cluster, what features it owns, how to set it up, and how to manage it.

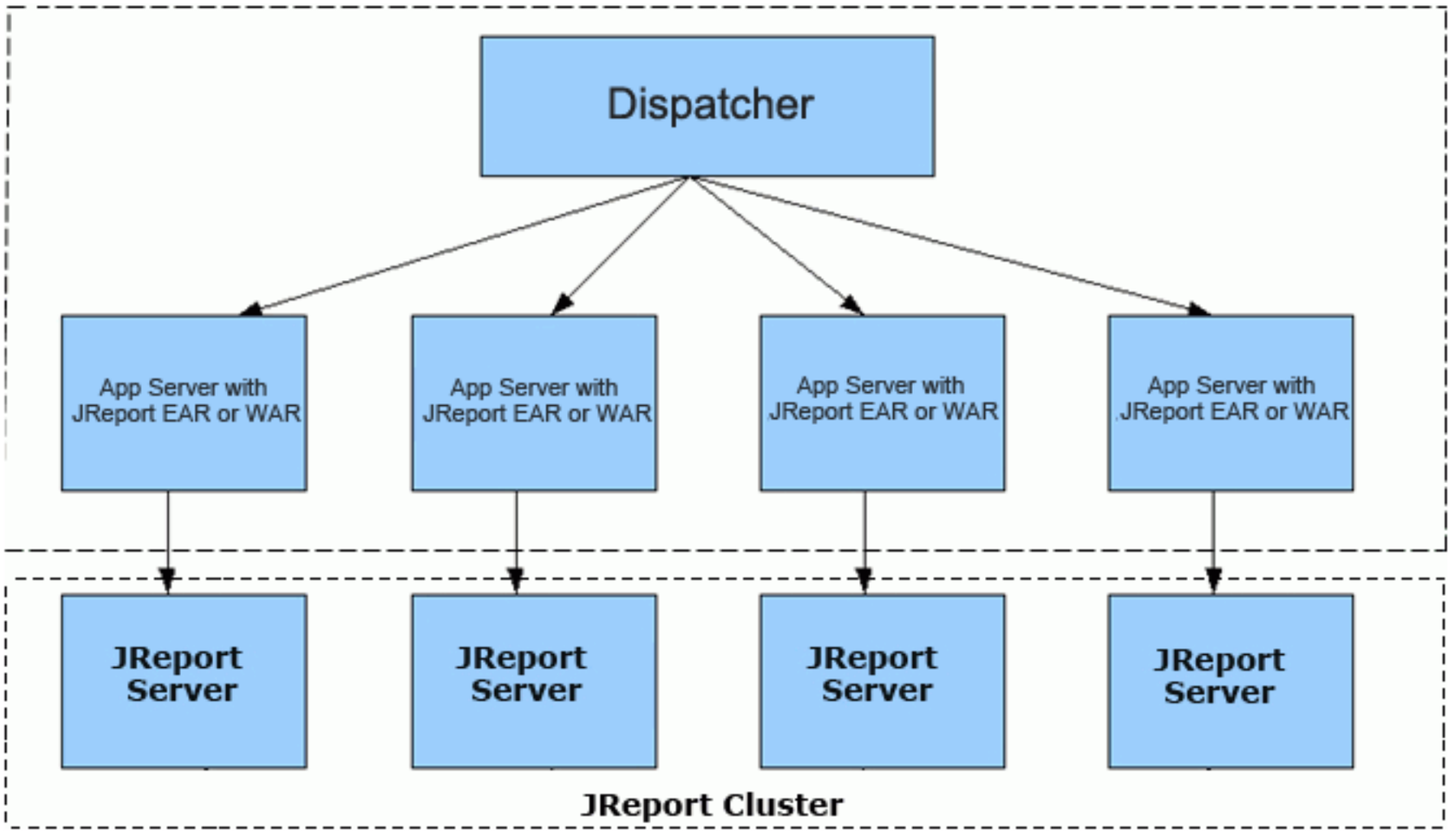
The following topics are covered:

- Cluster overview
- JReport Server Cluster main features
- Setting up and starting a JReport Server Cluster
- Managing a JReport Server Cluster
- Clustering JReport Server with Oracle WebLogic Server 12c (12.1.3)
- Dispatching RMI Server Pages requests in multiple server environment

Note: A JReport Server Cluster license is required in order to use this feature. If you do not have the license, contact your Jinfonet Software account manager to obtain it.

Cluster overview

There are many nodes (clustered servers) that play the same role in a JReport Server Cluster. The following is a diagram of the JReport Server Cluster infrastructure:



Every clustered server in this distributed cluster has the same responsibility. You can set each clustered server in a JReport Server Cluster by configuring its properties. The following list shows all the tasks each clustered server in the server cluster can complete.

	Business tasks			Administrative tasks			
	Run Reports	Submit Scheduled Tasks	Load-Balancing	Fail-over	Load-Balancing Administration	Security Administration	Resource Administration
Clustered Server	Y	Y	Y	Y	Y	Y	Y

Related topics:

- JReport Server Cluster

JReport Server Cluster main features

This section describes the main features of the JReport Server Cluster. Some of these features are also available through the API. With these features in mind, you will be better able to understand JReport Server Clusters and easily use them.

Administering security and resources

In a distributed cluster, you can accomplish all administrative tasks from any single node.

After logging onto the cluster from a clustered server as an administrator, certain security administrative tasks can be performed:

- Create, remove and edit users, groups, realms, protections and ACLs.
- Edit resource nodes and sub resource nodes. Add reachable virtual resource nodes.
- Customize the default page appearance for users.

Load balancing

Here are the benefits of deploying load balancing in a JReport Server Cluster.

- Automatically allocates tasks to suitable servers according to their current load and performance.
- Makes sure that all of the servers in the cluster are fully utilized.
- Automatically re-balances the network load when one server is added or removed.

For scheduled tasks and tasks submitted from interactive reports (reports that run on Page Report Studio, Web Report Studio, JDashboard or Visual Analysis), JReport adopts different load balancing mechanisms to enable the servers to work more effectively.

Load balancing for interactive reports

When running interactive reports in a cluster environment, JReport's built-in dispatcher or a user-customized dispatcher would be used to do load balancing.

JReport's built-in dispatcher is implemented on report level. It uses an internal load balancing algorithm which relies on computing resources such as memory and CPU usage to balance load. When an interactive report is opened, JReport will compute those resources and then allocate the most suitable server to the report. All later requests from the report will be performed on the server thereafter.

Customized dispatchers, which are difficult to implement, would be used quite rarely and only for special cases. That's because, when using customized dispatcher to do load balancing for interactive reports, firstly you need to provide all server nodes in the cluster in advance, and what's more, only session level dispatcher could be implemented.

Load balancing for scheduled tasks

Cluster Scheduler Lease

Every clustered server has a scheduler, and among the schedulers those with a lease are active schedulers. When the time of a scheduled task arrives, active schedulers compete and the winner gets to trigger the schedule. When dispatching tasks, the server which has the active scheduler will select a server according to load balancing algorithm and allocate the task to it.

By default, in a JReport Server Cluster all nodes of the cluster compete to become the active scheduler when the time of a scheduled task arrives. If the scheduled task has been bound with a trigger, then the node who gets the trigger becomes the active scheduler. The active scheduler for the task will then determine the server that will be asked to run the scheduled report based on load balancing. The Cluster Scheduler Lease option allows you to limit the number of servers competing for each scheduled report by setting a Cluster Scheduler Lease Active Count. As long as the count is less than the total number of nodes in the cluster, only the nodes holding a lease will compete to become the scheduler for the report that is ready to run. Depending on the number of scheduled reports you have, you may find that setting the Lease Active Count to 1 or 2 will provide more overall throughput on the system so the other nodes never have to be concerned about scheduled tasks.

There are two additional parameters that can be set:

- Cluster Scheduler Lease Valid Time will set the amount of time that the lease holder will continue to compete for scheduled tasks to run. The default value is 300 seconds.
- Cluster Schedule Lease Check Interval will set the amount of time between when other non-lease nodes will check to see if a lease is available to pick up. The default value is 30 seconds, that is, every 30 seconds all the other nodes will check to see if one of the lease semaphores is available to take. The number of semaphores is set by the Cluster Scheduler Lease Active Count.

Load detection

There is a JReport Server residing in each node of a JReport Server Cluster. The main factor that affects load balancing is the number of concurrent reports that are running on every JReport Server. In order to avoid heavy load, every member server in the cluster has been enabled to send the number of concurrently running reports on it to the other cluster nodes.

Built-in load balancing algorithms

JReport Server Cluster supports several algorithms for load balancing clustered servers. Configurable algorithms for load balancing clustered servers are:

- **Min-load (loadbalance.type=0)**
  The server that has the active scheduler will select the server which has the least number of currently running reports. If the local server is one of the qualified servers, it will be given higher priority.
- **Round Robin (loadbalance.type=1)**
  The server that has the active scheduler will select each server in sequence one by one until each has been allocated a report to run then will repeat the cycle. This is the default setting.
- **Weighted Min-load (loadbalance.type=2)**
  The server that has the active scheduler will select the server that has the least weighted current reports. If the local server is one of the qualified servers, it will be given higher priority.

$$\text{Weighted current reports} = \frac{\text{Number of currently running reports}}{\text{Performance Weight}}$$

  Performance weight is a positive floating point number that you set to each server in a cluster on any clustered server. Use JReport Administration page > Cluster > Weight panel and measure the performance of a typical report on each node of the cluster. The higher performance weight you set to a clustered server, the higher chance it may get selected by the server that holds the active scheduler during load balancing. See Configuring performance weight for how to set performance weight and how this algorithm works.

  If you do not set performance weight, by default the algorithm will work the same as Round Robin.
- **Random (loadbalance.type=3)**
  The server that holds the active scheduler will select the server randomly.

Failover

You can check the status of the clustered servers on JReport Server Monitor and notice the failure of any member server. If a member server is down, JReport cluster will remove it from the active clustered server list.

Member server failure

- **Effect on load balancing**
  When JReport cluster detects a failed member server, it will remove the member server from the active server list and will not schedule reporting tasks to that server any more. Load balancing will proceed on the remaining active servers.
- **Effect on incomplete tasks**
  When JReport cluster detects a failed clustered server, it will check the shared table for the list of incomplete tasks and will then reassign all incomplete tasks to other active servers using the load balancer.
- **Effect on completed tasks**
  JReport supports only report level recovery but not session level recovery. Once a report task is completed, it will be written to temporary storage for redirection to the requester. Failure after that will not be recovered.

Notifying of server down

If you have enabled the notifying of server down feature, when a member server crashes or is disconnected with the cluster, JReport cluster will send a notification e-mail to a specified address.

Distributed resource storage

In a pure distributed cluster the resource files are not stored in a central place. Each server has its own directories to be able to hold all types of resources including catalogs, templates, dashboards and report results. The sharing of resources among servers is achieved via a copying mechanism. More copies of resources reduce the possibility of resources being lost if a server goes down. However, making too many copies consumes system resources by making copies that are never used.

There are two copying scenarios. One is when a resource is saved, JReport will copy the resource to other randomly selected servers within the cluster, according to an argument: the number of copies allowed for this type of resource. The saved resource itself is regarded as a copy. That is, in the case when copy number is three, two other random servers will each be given a copy of the resource.

The other is on demand when a resource does not exist on the local server. JReport will search among the other servers for the resource and copy it to the local server.

The controllable number of copies argument is available to administrator users on the JReport Administration page > Cluster > Configuration panel. These types of resources support distribution: history resources which are resources saving into the versioning system, realm resources, CRD results that are cached report results, and memory storage related resources. You will be able to find the number of copies setting for each type.

The best number to use for each resource depends on two factors, how many points of failure do you want to tolerate and how often resources which are not local are requested by users. For example on a 10 node system, you might want to still run in case any two nodes go down. In this case, you would want the number of copies to be three. Any two nodes could go down and there would still be one with a copy of the resources. However, if there is a lot of on-demand requests that require copies you might find that a higher number actually provides better average performance for your users since they are not waiting for resources when they make on-demand requests.

Setting up and starting a JReport Server Cluster

This section introduces the steps for setting up a JReport Server Cluster and starting it. Here, it is assumed that you already have a general idea about the infrastructure of JReport Server Clusters and know the functions of the clustered servers. If this isn't the case, see Cluster overview and JReport Server Cluster.

Following are steps and examples for how to set up and start a JReport Server Cluster:

- Preparation
- Setting up and configuring a JReport Server Cluster
- Starting a JReport Server Cluster
- Example 1: Setting up a simple JReport Server Cluster
- Example 2: Setting up a JReport Server Cluster for a production environment

Preparation

To set up a JReport Server Cluster, you must determine the following factors:

- What is a JReport cluster?

  JReport cluster is based on JGroups and is a distributed cluster. All clustered servers in JReport cluster play the same role. You can add a new server to the existing cluster or shut down a server from the cluster any time.

- How many servers will be included in the cluster?

  The maximum number of servers is unlimited as long as you install JReport Server with the license key for cluster. Based on your expected load and protection from system failures you can create 2 or more nodes for your cluster. Often with multi-cpu and multi-core systems you will get better overall throughput having several nodes on a single server. Only by testing in your environment will you be able to find the number of nodes to give you the highest performance. Too few and resources will be under utilized and too many will cause thrashing and lower throughput.

- Whether to use the distributed storage feature in the cluster?

  In a distributed cluster the files may be stored on any node in the cluster. You can set how many copies will be made in the cluster and if you need to access the files from another node, JReport cluster will copy them to this node from the node it is stored on. As a result, you can access your required files from anywhere in the cluster. If the copy number is 0, then it means every node of the cluster will get a copy.

  The tradeoff is overall system performance versus individual user performance when a user requests a report result. If you set Number of Copies to 0, the system will copy every resource file to every node, slowing overall throughput considerably. However, setting Number of Copies to 1 will keep a single copy just on the node where it was created which provides maximum system throughput but when a user requests a resource which is not on his node he then needs to wait for it to be copied before he can view it. If the node goes down though then the resource is unreachable. A setting of 2 is the default which allows for failover if a node goes down but just does a single copy.

- Whether to use the default Derby DBMS or use your own DBMS?

  JReport includes the Apache Derby DBMS for the server data such as resources and users, groups and roles and a lot of other information. By default each installed node creates it's own database in `<install_root>/derby`. In order to use a JReport cluster, all nodes must use the same database. Select one of the nodes to manage the Derby DBMS and ensure that all the other nodes point to this same instance. For an example, review the sample configuration Case 1.

  Another option to consider is using your own DBMS for the server database. If you already have a reliable DBMS which is already being backed up and provides the reliability you need such as MySQL or Oracle we recommend you change the system DBMS to use your own managed DBMS rather than maintain a separate one for JReport. For information on how to configure JReport to use a different system DBMS, refer to Configure the server database.

Setting up and configuring a JReport Server Cluster

Before setting up a JReport Server Cluster, first you need to make the time difference between the target computers that will join the cluster as small as possible (the time difference between the target computers should be within one minute), and make sure all servers in the cluster will be set up on the same architecture and operating system.

A JReport Server Cluster can be set up either during the JReport Server installation or after the servers have been installed using the appropriate license key for cluster.

Creating a JReport Server Cluster during installation

It is recommended that you create a JReport Server Cluster during the installation of servers. This is the easiest way for you to set up a cluster.

To create a JReport Server Cluster during installation:

1. Run the JReport Server installation file to install JReport Server with the Installation Wizard.
2. Specifies your User ID. In the License Key text field, use the cluster enabled license key.
3. When choosing the installation type, choose **Custom Installation for Standalone Server**. Then specify the installation directory for JReport Server.
4. In the Service tab of the Configure System Environment screen, check the **Network Address At** option and type in the IP address of the server.
5. In the Cluster tab, specify a cluster name in the Cluster Name text field. You can either make the server join an existing cluster or specify a new cluster name to build another cluster.
6. Uncheck the **Disable Cluster** option.
7. From the Load Balancer Type drop-down list, select the algorithm for load balancing clustered servers.
8. Check the **Cluster Scheduler Lease** option to enable lease for the cluster, then set the active count, valid time and check interval for the cluster scheduler lease. If you don't enable the lease option, all clustered servers in the cluster will compete for a chance to trigger scheduled tasks which could lower overall system throughput. Read main features for additional information.
9. Specify values for Cluster Storage History Number of Copies, Cluster Storage Realm Number of Copies and Cluster Storage CRD Result Number of Copies. If you are using shared disk resources for any of these directories you should set the value to 1. The default value is 2 which means make one copy plus the original. This allows any one node to go down and the system will still be able to find all resources. If you want to allow 2 simultaneous failures, set the number of copies to 3.
10. Specify the value for Cluster Memory Storage Number of Copies to set how many memory copies will be shared in the cluster.
11. Check **Notify via E-mail When a Server Is Down** if you need to notify somebody via e-mail when a server in the cluster is down, then in the E-mail Address text field, input the e-mail addresses of the people to whom you want to send a notification e-mail.
12. Set the Properties, Realm, Resource, History, and Temporary Directories if necessary. If they are not specified, default directories will be used.
13. Type the IP address or host name of the server in the Server's RMI Host text field, and type the port number in the Server's RMI Port text field.
14. Go on with the installation steps to complete the installation.
15. Install another server you want to join the cluster and repeat the above steps to configure its cluster settings.

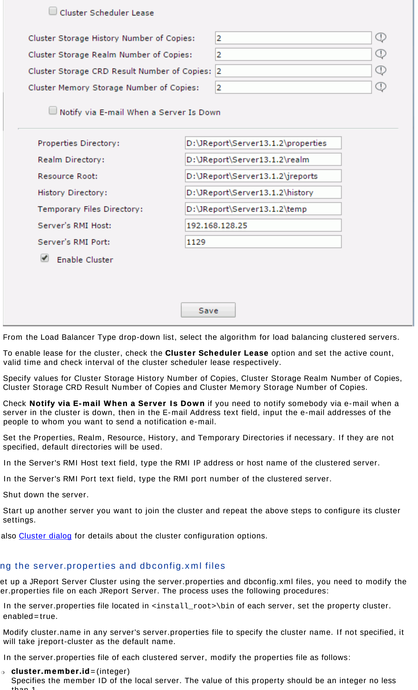
    Since JReport Cluster uses the same server DBMS, you need to make the system database in the JDBC URL text field point to the same DBMS as the previous server. This requires that you replace localhost with the IP address of the first cluster node that you installed, for example, `jdbc:derby://IP:1527/`.

Creating a JReport Server Cluster after installation

After the servers have been installed using the appropriate license key for cluster, you can configure the clustered servers either by using the JReport Administration page or the server.properties and dbconfig.xml files.

Using the JReport Administration page

1. Start up the server that hasn't been enabled for cluster.
2. Log onto the JReport Administration page, and click **Cluster** > **Configuration** on the system toolbar to show the panel.
3. In the Cluster Name text field, specify a name for the cluster. You can either make the server join an existing cluster or specify a new cluster name to build another cluster.
4. Check the **Enable Cluster** option.
5. Click **Save** to enable the cluster.
6. A cluster member ID will be generated automatically for the server. But if you need to modify it, go to server.properties file located in `<install_root>\bin` and set the property cluster.member.id.
7. Go to the JReport Administration > Data page, configure the databases to make sure they point to the database that the server will use. For details, see Configuring the server database.
8. Restart the server you have enabled with cluster, then log onto the JReport Administration page, go to the **Cluster** > **Configuration** panel.



9. From the Load Balancer Type drop-down list, select the algorithm for load balancing clustered servers.
10. To enable lease for the cluster, check the **Cluster Scheduler Lease** option and set the active count, valid time and check interval of the cluster scheduler lease respectively.
11. Specify values for Cluster Storage History Number of Copies, Cluster Storage Realm Number of Copies, Cluster Storage CRD Result Number of Copies and Cluster Memory Storage Number of Copies.
12. Check **Notify via E-mail When a Server Is Down** if you need to notify somebody via e-mail when a server in the cluster is down, then in the E-mail Address text field, input the e-mail addresses of the people to whom you want to send a notification e-mail.
13. Set the Properties, Realm, Resource, History, and Temporary Directories if necessary. If they are not specified, default directories will be used.
14. In the Server's RMI Host text field, type the RMI IP address or host name of the clustered server.
15. In the Server's RMI Port text field, type the RMI port number of the clustered server.
16. Shut down the server.
17. Start up another server you want to join the cluster and repeat the above steps to configure its cluster settings.

See also Cluster dialog for details about the cluster configuration options.

Using the server.properties and dbconfig.xml files

To set up a JReport Server Cluster using the server.properties and dbconfig.xml files, you need to modify the server.properties file on each JReport Server. The process uses the following procedures:

1. In the server.properties file located in `<install_root>\bin` of each server, set the property cluster.enabled=true.
2. Modify cluster.name in any server's server.properties file to specify the cluster name. If not specified, it will take jreport-cluster as the default name.
3. In the server.properties file of each clustered server, modify the properties file as follows:
   - **cluster.member.id**=(integer)
     Specifies the member ID of the local server. The value of this property should be an integer no less than 1.
   - **server.rmi.host**=localhost IP address
     Specifies the RMI IP address or host name of the local server.
   - **server.rmi.port=1129**
     Specifies the RMI port number of the local server.
4. Specify the paths of the directories on the physical disk, such as properties directory, realm directory, resource root, history directory, and temporary files directory. The directories on each clustered server should point to a different physical disk.

   Add the following properties to the server.properties file:
   - **resource.share.temp.dir=**
     Specifies the directories for storing temporary files.
   - **resource.share.hist.dir=**
     Specifies the directories for storing all versions of report results in the cluster.
   - **resource.share.realm.dir=**
     Specifies the directories of the realm files.
   - **resource.share.properties.dir=**
     Specifies the directories of the properties files.
   - **resource.root=**
     Specifies the directories of the JReport demo reports.
5. Modify dbconfig.xml in each server's `<install_root>\bin`, and make sure that the system database and realm database all server nodes use point to the same DBMS. For details, see Configuring the server database.

Notes:

- The databases that can be used as the server database in a JReport cluster must be able to support the lock function, therefore, you cannot use HSQLDB as the server's database in a JReport cluster because HSQLDB doesn't support the function.
- If you have upgraded your JReport Server which is cluster enabled and uses HSQLDB as the server database from versions prior to V9 to the current version, in order to make the server work normally, you need to disable cluster on the server by adding the line server.cluster.enable=false to the server.properties file.
- If you set up a JReport cluster on one computer, you need to make sure that the settings of HTTP Port, Administration Port, and Server's RMI Port on each cluster node be different from those on any other nodes.
- It is strongly recommended that you do not change the auto generated cluster member ID in a JReport cluster, because distributed storage uses the member ID to recognize on which node the physical files are stored.
- If you want to use resources from folder real paths in JReport cluster, you should first specify a disk in your file system and make sure that every node in the cluster has the same disk name mapping to the same physical location.
- Properties such as cluster.name, cluster.member.id, resource.share.properties.dir, resource.share.realm.dir, resource.root, resource.share.hist.dir, resource.share.temp.dir, server.rmi.host, server.rmi.port, cluster.enable, log.config.filename, log.config.update, and log.config.update.interval that created in the server.properties file and their corresponding UI options on the JReport Administration page need to be configured respectively for each cluster node. However, properties that are now saved in the database should be configured for all the server nodes in the cluster. These properties are:
  - The following properties listed before in the server.properties file: cluster.enable_notify_server_down, cluster.notify_server_down_address, cluster.scheduler.lease.active_count, cluster.scheduler.lease.check_interval, cluster.scheduler.lease.enabled, cluster.scheduler.lease.valid_time, cluster.share_memory.node_number, cluster.storage.crd_result.copy_number, cluster.storage.history.copy_number, cluster.storage.realm.copy_number, loadbalance.custom_class, loadbalance.type, server.autocache.enabled, server.autocache.expired.time, server.autocache.max.disk.usage, server.autocache.never.expire, server.completed.max_count, server.crd.memory.usage, server.realm.active, server.security, server.version.from.temp.
  - All properties in the mailconfig.properties file in `<install_root>\bin`.
  - All properties in the clusterWeight.properties file in the system database.
  - All properties in the LDAP configuration XML file LDAPProperties.xml in `<install_root>\properties`.
- No sub folders should be created under the realm directory because it may create a realm when server is started.
- To use the old result files in `<reportrhome>\history`, install the new JReport Server to the same directory with a new license key, overriding the existing program files.
- In order to fax report result successfully in a cluster, you need to configure the fax settings for each clustered server respectively.
- If the content of the resource root directory jreports is removed, the Public Reports folder in the resource tree will be empty.

Starting a JReport Server Cluster

To start a JReport Server Cluster, start the servers you have configured for the cluster one by one. If you are using the default Derby DBMS, be sure to start Derby first by running startNetworkServer.bat/sh in the `<install_root>\derby\bin` directory on the server containing the server DBMS.

Notes:

- If you are using multiple IP address on a clustered server, you need to add `-Djgroups.bind_addr=IP address` at which JReport cluster can work properly to its JRServer.bat file located in `<install_root>\bin` to make sure the server can be started successfully.
- If you have two JReport clusters with the same cluster name in a network segment, although the two clusters are pointing to different databases, only the one started earlier can work successfully.

Example 1: Setting up a simple JReport Server Cluster

This example demonstrates how to configure a simple JReport Server Cluster by modifying the configuration options on the JReport Administration page on each JReport Server.

Example description:

- Set up a simple JReport Server Cluster using the JReport Administration page. Assume that JReport Server Monitor has been installed on your computer.
- The cluster consists of two copies of JReport Server on one computer.
- The cluster uses one server DBMS.
- The cluster uses shared directories for resources so no resource copies are required.

Take the following steps to set up the cluster:

1. Install the two JReport Servers respectively to `C:\JReport\Server1` and `C:\JReport\Server2` using the cluster enabled license key.
2. Launch the JReport Server installed to `C:\JReport\Server2`.
3. Log onto the JReport Administration page of Server2, click **Configuration** > **Service** on the system toolbar, then set Port, Dashboard Port, and Administration Port respectively to **8885**, **8884** and **8883** to make them different from those of Server1. You may use any port numbers which are available on your system.
4. Click **Cluster** > **Configuration** on the system toolbar.
5. In the Configuration panel, specify a cluster name and check the **Enable Cluster** option, then click **Save** to enable the cluster.
6. Restart the JReport Server installed to `C:\JReport\Server2`.
7. Log onto the JReport Administration page of Server2 using the administration port 8883 set in Step 3 (`http://localhost:8883`).
8. Go to the **Cluster** > **Configuration** panel, leave the Load Balancer Type as Round Robin.
9. Check the **Cluster Scheduler Lease** option to enable lease for the cluster, then set the active count, valid time and check interval for the cluster scheduler lease.
10. Change the Cluster Storage History, Realm and CRD Result Number of Copies to **1**. We will just use one resource directory.
11. Keep the default value 2 for Cluster Memory Storage Number of Copies, thus 2 memory copies will be shared in the cluster.
12. Check the **Notify via E-mail When a Server Is Down** option, and in the E-mail Address text field, input the e-mail addresses of the people to whom you want to send a notification e-mail.
13. Keep the default values for Properties Directory, Realm Directory, Resource Root, History Directory and Temporary Files Directory.
14. Type the IP address or host name of Server2 in the Server's RMI Host text field, and type the port number in the Server's RMI Port text field.

    The port is the RMI port of the clustered server. The default port number is 1129. If there are two or more JReport Servers started on one machine, the RMI port number of each clustered server must be changed to a unique one, in order to avoid port conflicts.

    In this example, the port number is changed to 1130, since the other server will use the default port number 1129.
15. Click **Save** to accept all the changes, then shut down the server.
16. Edit `C:\JReport\Server2\bin\dbconfig.xml` and remove the lines with auto-start-derbyservice. We only want Server1 to start the server DBMS since we will always use the Server1 database.
17. Launch the JReport Server installed to `C:\JReport\Server1`, log onto the JReport Administration page (`http://localhost:8889`), then click **Cluster** > **Configuration** on the system toolbar.
18. Use the same cluster name as Server2, thus making Server1 join the existing cluster. Check the **Enable Cluster** option, then click **Save** to enable the cluster.
19. Restart the JReport Server installed to `C:\JReport\Server1`, and then log onto the JReport Administration page.
20. Go to the **Cluster** > **Configuration** panel, configure Server's RMI Host and Server's RMI Port. Remember to keep Server's RMI Port to its default value 1129.
21. Upon finish, click **Save** to accept all settings and shut down Server1.
22. Copy rmi.auth in `C:\JReport\Server1\bin` to `C:\JReport\Server2\bin`. This allows RMI to be authorized between the two systems.
23. Edit server.properties in `C:\JReport\Server2\bin` and remove cluster.member.id. It will be recreated when you restart Server2 with a unique number.
24. Start the server Derby DBMS service by double-clicking the startNetworkServer.bat file in `C:\JReport\Server1\derby\bin`.
25. Restart Server1 and Server2. It doesn't matter which one you started first.
26. Start JReport Server Monitor and check the cluster on the JReport Monitor page.
27. Access the JReport Console page of the first server using port 8888 as an administrator, and then submit a scheduled task. In the Scheduled tab, you will see the newly scheduled task.
28. Log onto the JReport Administration page of the second server and create a new user Tom in the **Security > User** panel. Access the JReport Console page of the second server using port 8883 as Tom, and then submit another scheduled task.

Notes:

- When you schedule to publish a report on a clustered server to disk, you should first specify a disk in your file system and make sure that every node in the cluster has the same disk name mapping to the same physical location. Then the publishing result will be directly saved to the disk you specify.
- You can only view scheduled tasks that you have submitted.
- From the JReport Console page of the clustered servers, you can only view completed tasks that you have submitted.
- If there are more than two clustered servers in the cluster, then after you shut down one server, all the scheduled tasks running on this server will be run on other servers.

Example 2: Setting up a JReport Server Cluster for a production environment

This example demonstrates how to set up a JReport Server Cluster on Unix/Linux by configuring the cluster UI on each JReport Server. There will be three computers in the cluster. They are node1 (IP address: 192.168.0.1), node2 (IP address: 192.168.0.2) and node3 (IP address: 192.168.0.3). All JReport Servers in the cluster use Apache Derby as the server system database.

Take the following steps to set up the cluster:

1. Make the time difference between the target computers be within one minute.
2. Install JReport Server on each of the three nodes in directories `/home/JReport/Server1`, `/home/JReport/Server2` and `/home/JReport/Server3` respectively using the appropriate license key for the cluster, and install JReport Server Monitor on one of the nodes.
3. Launch the JReport Server installed to `/home/JReport/Server1` on node1.
4. Log onto the JReport Administration page and click **Cluster** > **Configuration** on the system toolbar. In the Configuration panel, specify a cluster name and check the **Enable Cluster** option, then click **Save** to enable the cluster.
5. Restart the JReport Server on node1, and go to the JReport Administration page > Cluster > Configuration panel.
6. Leave the Load Balancer Type as Round Robin.
7. Check the **Cluster Scheduler Lease** option to enable scheduler leases for the cluster, then set the active count, valid time and check interval for the cluster scheduler lease respectively. The defaults of 2 active schedules with a time of 300 seconds and check interval of 30 seconds are a good starting point.
8. Keep the cluster storage history, realm and CRD result number of copies to 2. This means that each resource will be copied to 2 of the 3 servers so there will be no single point of failure for the resources.
9. Keep the default value 2 for Cluster Memory Storage Number of Copies, thus 2 memory copies will be shared in the cluster.
10. Check the **Notify via E-mail When a Server Is Down** option, and in the E-mail Address text field, input the e-mail addresses of the people to whom you want to send a notification e-mail.
11. Keep the default values for Properties Directory, Realm Directory, Resource Root, History Directory and Temporary Files Directory.
12. In the Server's RMI Host text field, type the IP address or host name of Server1 as **192.168.0.1**. Type the port number in the Server's RMI Port text field as **1129**.
13. Click **Save** to accept all the changes.
14. Go to the **Data** > **System DB** panel of Server1, in the Configuration tab, copy the URL in the URL text field to a temporary file, and go to the Realm DB panel to copy the URL using the same way.

    In order to use a JReport Server Cluster, all nodes in the cluster must use the same database. In this case, all three nodes will use the database Server1 uses. This ensures that all servers in the cluster share a single DBMS instance.
15. Shut down the server.
16. Launch the JReport Server installed to `/home/JReport/Server2` on node2.
17. Log onto the JReport Administration page and click **Cluster** > **Configuration** on the system toolbar. In the Configuration panel, use the same cluster name as Server1, thus making Server2 join the existing cluster. Check the **Enable Cluster** option, then click **Save** to enable the cluster.
18. Restart the JReport Server on node2. Go to the JReport Administration page > Data > System DB/Realm DB, make the system database and realm database in the URL text field the same as those of node1.
19. Go to the **Cluster > Configuration** panel.
20. In the Server's RMI Host text field, type the IP address or host name of Server2 as **192.168.0.2**. Type the port number in the Server's RMI Port text field as **1129**.
21. Click **Save** to accept all the changes, then shut down the server.
22. Copy rmi.auth in `/home/JReport/Server1/bin` to `/home/JReport/Server2/bin`. This allows RMI to be authorized between node1 and node2.
23. Launch the JReport Server installed to `/home/JReport/Server3` on node3. Just like what we did with node2, enable cluster and make node3 join the existing cluster too.
24. Restart the JReport Server on node3. Go to the JReport Administration page > Data > System DB/Realm DB, make the system database and realm database in the URL text field the same as those of node1.
25. Go to the **Cluster > Configuration** panel.
26. In the Server's RMI Host text field, type the IP address or host name of Server3 as **192.168.0.3**. Type the port number in the Server's RMI Port text field as **1129**.
27. Click **Save** to accept all the changes, then shut down the server.
28. Just as we did with node2, copy rmi.auth in `/home/JReport/Server1/bin` to `/home/JReport/Server3/bin`.
29. Start the server Derby DBMS service by running the startNetworkServer.sh file in `/home/JReport/Server1/derby/bin`.
30. Launch the server on node1. In the Command Prompt window, you will see the following information:

    ```
    JReport Server is ready for service.
    ```
31. Similarly, launch the server on node2 and node3.
32. Start JReport Server Monitor and go to the JReport Monitor page to track the cluster.

Managing a JReport Server Cluster

The JReport Server Cluster page allows you to enable a cluster and configure and evaluate the performance weight of the servers in the cluster. On this page, you can:

- Enable and set up a cluster.
- Configure the clustered servers.
- Set the shared parameters of the clustered servers.
- Evaluate the performance weight of the clustered servers.

On the JReport Administration page, you can also administer the servers in a cluster. In this section, JReport Server Cluster administration is described in the following topics:

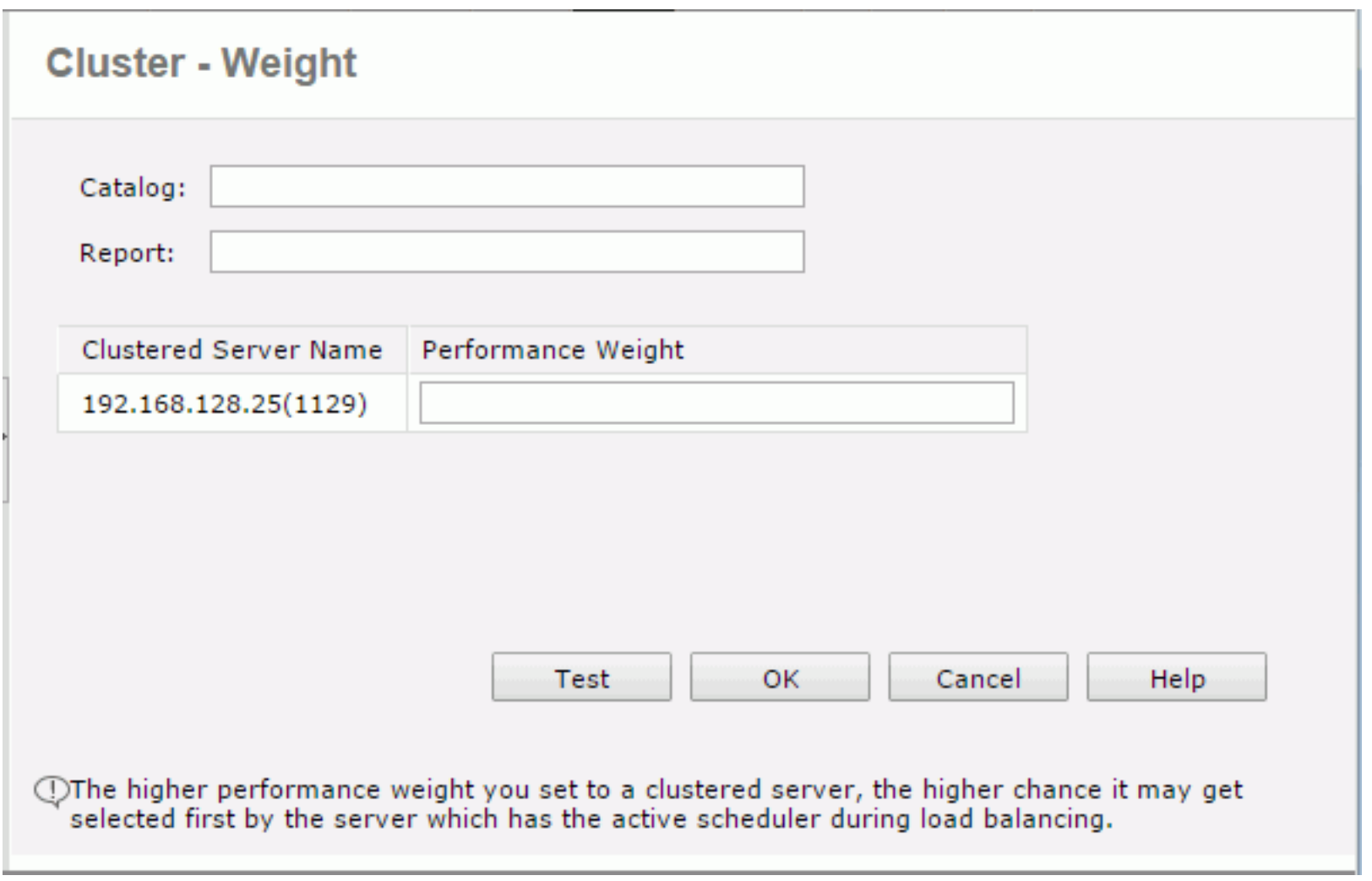
- Configuring performance weight
- Balancing the server load
- Monitoring clustered servers

Configuring performance weight

If you have chosen the Weighted Min-load (loadbalance.type=2) algorithm for load balancing, you will have to configure a performance weight for each clustered server in the cluster. The higher performance weight you set to a clustered server, the higher chance it may get selected by the server that has the active scheduler during load balancing.

To configure performance weight:

1. Start a clustered server in the cluster, on its JReport Administration page, click **Cluster** > **Weight** to show the panel.



2. Specify a weight value for each clustered server manually. Performance weight is a positive float number.
3. Click **OK** to save the weight values.
4. If you want to test each clustered server's performance weight value at current time, specify a catalog and a report that will be used for the testing in the Catalog and Report text fields and then click the **Test** button.

The following are two examples for how Least Weighted Current Reports algorithm works:

Example 1 - when there are free servers:

Active Servers	ServerA	ServerB	ServerC	Comments
Is local server	TRUE	FALSE	FALSE	
Maximum concurrent reports	8	Unlimited	5	
Number of currently running reports	6	6	5	
Performance weight	10	10	10	
Calculation				
Weighted current reports	0.6	0.6	0.5	
Is free	TRUE	TRUE	FALSE	Current < MaxConcurrent, or MaxConcurrent is unlimited
Candidate servers	YES	YES		Select from free servers
Candidate servers	YES	YES		Select servers which have the least Weighted current reports
Selected server	YES			Local server has higher priority

Example 2 - when there are no free servers:

Active Servers	ServerA	ServerB	ServerC	Comments
Is local server	TRUE	FALSE	FALSE	
Maximum concurrent reports	10	10	10	
Number of currently running reports	10	10	10	
Performance weight	4	5	8	
Calculation				
Weighted current reports	2.5	2	1.25	
Is free	FALSE	FALSE	FALSE	Current < MaxConcurrent, or MaxConcurrent is unlimited
Candidate servers	YES	YES	YES	Selects from all servers when all servers are full
Candidate servers			YES	Select servers which have the least Weighted current reports
Selected server			YES	Select ServerC

Balancing the server load

In a cluster environment, JReport Server provides a load balancing mechanism which enables the server to work more effectively.

Load balancing process

1. When the time of a scheduled task arrives, active schedulers compete and the winner gets to trigger the schedule.
2. The server that has the active scheduler selects a server in the cluster according to the load balancing algorithm specified which can either be a built-in one or a customized one, and then sends the task to the selected server.

Tip: You can also directly specify a server in a cluster to perform a scheduled task instead of using load balancing. To do this, first make sure that the Identify Server Preference option is enabled in the Profile dialog (**Profile** > **Customize Server Preferences** > **Advanced** > **Identify Server Preference**), and then use the Specify a preferred server to run the task option in the General tab of the Schedule dialog to specify a server manually.

Customized load balancing algorithm based on API

You can write your own load balancing algorithm based on the API included in JReport Server. Note that if you create a load balancing algorithm with the API, it will take effect in place of other built-in load balance algorithms you have set.

A demo DemoLoadBalancer.java has been provided to illustrate how to customize load balancing using APIs. You can find it in `<install_root>\help\samples\APICluster`.

Assuming that you have several clustered servers.

Take the following steps:

1. Compile DemoLoadBalancer.java to generate the class file as follows (when compiling DemoLoadBalancer.java, you need to add JRESServlets.jar to the class path):

   ```
   javac -classpath <install_root>\lib\JRESServlets.jar DemoLoadBalancer.java
   ```

2. Add DemoLoadBalancer.class to the class path of setenv.bat in the ADDCLASSPATH variable.
3. Add the parameter `-Dloadbalance.custom_class=DemoLoadBalancer` to the server's startup file JRServer.bat which locates in `<install_root>\bin`. For example:

   ```
   "%JAVAHOME%\bin\java.exe" -Dloadbalance.custom_class=DemoLoadBalancer "-Dinstall.
   root=%REPORTHOME%" ...
   ```

4. Launch JRServer.bat, the customized loadbalancer DemoLoadBalancer will then be applied.
5. Submit some tasks for running. You will now find that these tasks are allocated to the clustered servers based on the DemoLoadBalancer code.

Reference: For more details, see the jet.server.api.cluster.LoadBalancer interface in JReport Javadoc located in `<install_root>\help\api`.

Notes:

- You can choose the load balancing type by setting the API method setLoadBalanceType() at jet.server.api.admin.ClusterAdminService. For example, setting the API method as setLoadBalanceType(0), setLoadBalanceType(1), setLoadBalanceType(2), and setLoadBalanceType(3) means respectively the algorithm Least Current Reports (Min-load), Round Robin, Least Weighted Current Reports (Weighted Min-load) and Random will be chosen.
- For the load balancing algorithms: the server that holds the active scheduler selects from the servers with the number of concurrently running reports less than maximum number first. However, if all servers are full, it will select from all of them.

Monitoring clustered servers

JReport Server Monitor is a standalone web-based application used for monitoring the overall performance of JReport Server. JReport Server Monitor should be used together with JReport Server.

JReport Server Monitor contains the following main features:

- Inspects the status of JReport Server.
- Shows server performance statistics in Graph/Text mode.
- Maintains JReport Server.
- Creates profiling reports: performance reports and statistics reports.

Before you can use JReport Server Monitor to monitor servers in a cluster, the following steps must be taken:

1. Download and install JReport Server Monitor.
2. Modify the server.properties file in `<monitor_install_root>\bin` to configure the IP address and port information of one clustered server.
3. Copy rmi.auth from `<server_install_root>\bin` of the clustered server whose IP address and port information you modified in the last step to `<monitor_install_root>\bin`, or remove rmi.auth from `<server_install_root>\bin` of this clustered server.
4. Start JReport Server.
5. Launch MonitorServer.bat in `<monitor_install_root>\bin` to start JReport Server Monitor.
6. Access JReport Server Monitor using `http://monitorhost:monitorport (default 8848)`, or by clicking the Monitor link on the JReport Administration page.

Note: The Monitor link will not be displayed on the JReport Administration page when the web.monitor.link.enable property in the server.properties file in `<server_install_root>\bin` is set to false. You can specify the monitor port by setting monitor.jmx.htmladaptor.port in server.properties.

Related topics:

- Monitoring JReport Server

Clustering JReport Server with Oracle WebLogic Server 12c (12.1.3)

This section describes how to configure a WebLogic 12c cluster and then integrate JReport Server in the cluster.

Assuming that the server cluster contains an admin server named adminNew (port 7001), a proxy server named proxy, and two managed servers named manage1 and manage2. Each clustered computer should have a unique IP address or host name.

Installing Oracle WebLogic Server 12c (12.1.3)

1. Run the installation file fmw_12.1.3.0.0_wls.jar downloaded from Oracle website to start the Oracle Fusion Middleware 12c WebLogic Server and Coherence Installation wizard.
2. Click **Next**. In the Installation Type screen, choose **Coherence Installation**.
3. Keep clicking **Next** till the Installation Complete screen, then click **Finish** to finish the installation.

Configuring an Oracle WebLogic cluster

1. Start the Configuration Wizard. On Windows you can start it from the Start menu by selecting **Start** > **All Programs** > **Oracle** > **OracleHome** > **WebLogic Server 12c (12.1.3)** > **Tools** > **Configuration Wizard**.
2. In the Create Domain page, select **Create a new domain** and then click **Next**.
3. In the Templates page, check **Create Domain Using Product Templates** and choose **WebLogic Coherence Cluster Extension - 12.1.3.0 [wlserver]** from the available templates list, and then click **Next**.
4. In the Administrator Account page, enter the name and password, and then click **Next**.
5. In the Domain Mode and JDK page, choose **Production** for Domain Mode and **Oracle HotSpot 1.7.0_40 D:\Java\JDK17~1.0_4** for JDK. Click **Next**.
6. In the Advanced Configuration page, keep the three advanced configuration options selected, then click **Next**.
7. In the Administration Server page, enter the IP address of your machine in the Listen Address field. Change or accept the default settings for **Server Name** and **Listen Port**. Then click **Next**.
8. In the Node Manager page, choose **Per Domain** for Node Manager Type. and then click **Next**.
9. In the Managed Servers page, click **Add** to add two managed servers and a proxy server as follows:

Name	Listen Address	Listen port
manage1	127.0.0.114	7006
manage2	127.0.0.131	7007
proxy	127.0.0.114	7008

   Note: Keep in mind that each server instance in your cluster, including the admin server, proxy server and all managed servers, must have a unique Listen Address and Listen Port. If your managed server has the same Listen Address as the admin server, change the default values shown for the listen ports in this page, and enter the same Listen Address as you assigned to the Admin Server. After assigning a unique Listen Address and Listen Port combination, click the **Add** button to create another managed server.

10. In the Clusters page, add a cluster named **Cluster_28**. Then click **Next**.
11. In the Assign Servers to Clusters page, highlight the managed servers and the proxy server in the left column, and then use the right arrow button to assign the servers to the cluster listed in the right column. Then click **Next**.
12. In the Coherence Clusters page, set the Cluster Name as **defaultCoherenceCluster** and Unicast Listen Port as **0**, and then click **Next**.
13. In the Machines page, add two machines as follows:

Name	Node Manager Listen Address	Node Manager Listen Port
Machine_zjh	127.0.0.114	5555
Machine_zwm	127.0.0.131	5556

14. Assign servers to machines. Assign the admin server, server manage1 and the proxy server to the machine Machine_zjh.
15. Review your configuration specifications and configure the domain. Set Domain Name to **mydomain28** and Location to `C:\Oracle12c\Middleware\Oracle_Home\user_projects\domains\mydomain28`. Then click **Create**.
16. Start the servers on machine Machine_zjh:

    Start the admin server on the admin server:

    `C:\Oracle12c\Middleware\Oracle_Home\user_projects\domains\mydomain28\bin\startWebLogic.cmd`.

    Start the node manager:

    `C:\Oracle12c\Middleware\Oracle_Home\user_projects\domains\mydomain28\bin\startNodeManager.cmd`.

    Start and log onto the proxy server on the proxy server:

    `C:\Oracle12c\Middleware\Oracle_Home\user_projects\domains\mydomain28\bin\startManagedWebLogic.cmd proxy http://127.0.0.114:7001`.

    Start and log onto the manage1 server on manage1 server:

    `C:\Oracle12c\Middleware\Oracle_Home\user_projects\domains\mydomain28\bin\startManagedWebLogic.cmd manage1 http://127.0.0.114:7001`.

17. Now let's configure another managed server. Repeat step 1 to 8 for manage2.
18. Skip the Managed Servers page, the Clusters page and the Assign Servers to Clusters page. In the Coherence Clusters page, set the Cluster Name as **defaultCoherenceCluster** and Unicast Listen Port as **0**, and then click **Next**.
19. Skip the Machines page. In the Assign Servers to Machines page, assign server manage2 to the machine Machine_zwm. Then click **Next**.
20. In the configuration Summary page, also set Domain Name to **mydomain28** and Location to `C:\Oracle12c\Middleware\Oracle_Home\user_projects\domains\mydomain28`. Then click **Create**.
21. Start the servers on machine Machine_zwm:

    Start the admin server on the admin server:

    `C:\Oracle12c\Middleware\Oracle_Home\user_projects\domains\mydomain28\bin\startWebLogic.cmd`.

    Start the node manager:

    `C:\Oracle12c\Middleware\Oracle_Home\user_projects\domains\mydomain28\bin\startNodeManager.cmd`.

    Start and log onto the manage2 server on manage2 server:

    `C:\Oracle12c\Middleware\Oracle_Home\user_projects\domains\mydomain28\bin\startManagedWebLogic.cmd manage2 http://127.0.0.114:7001`.

Deploying JReport to the Oracle WebLogic cluster

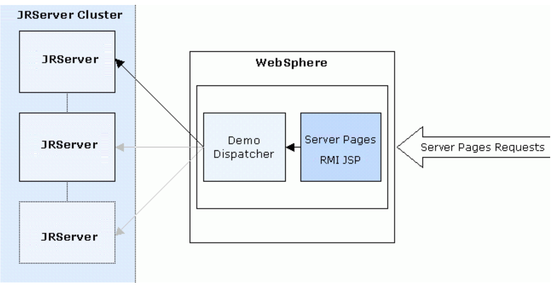
1. Start WebLogic by running startWeblogic.cmd in `C:\Oracle12c\Middleware\Oracle_Home\user_projects\domains\mydomain28\bin`.
2. Access the WebLogic Server Administrative Console 12c by using URL `http://127.0.0.114:7001/console/`, where 127.0.0.114 is host name or IP address, and 7001 is the port number.
3. Go to **Home** > **Summary of Environment** > **Summary of Servers** > **Summary of Deployments**. In the Available targets for jreport section, choose the cluster named **Cluster_28** and check **Part of the cluster**, then choose the two managed servers **manage1** and **manage2**. Click **Finish** to finish selecting deployment targets.
4. Go to **Home** > **Summary of Environment** > **Summary of Servers** > **Summary of Deployments** > **jreport**, click the **Save** button to save the settings for jreport. When you see the message Settings updated successfully, click the **Activate Changes** button in the Change Center panel on the left.
5. Continue to go to **Home** > **Summary of Environment** > **Summary of Servers** > **Summary of Deployments** > **jreport** > **Summary of Deployments**. In the Deployments section of Control panel, click the **Start** drop-down list and choose **Servicing all requests**. Click the **Lock & Edit** button in the Change Center panel on the left.
6. Wait until the deployments state of jreport changes to Active.
7. Start JReport Server using the URL `http://127.0.0.114:7008/jreport`. 127.0.0.114 is host name or IP address for the proxy server, and 7008 is its port number.

Dispatching RMI Server Pages requests in multiple server environment

You can dispatch RMI Server Pages requests in multiple server environment, which includes JReport clustered and non-clustered server environment.

Sample solution: dispatch RMI Server Pages requests in clustered server environment using customized dispatcher

This sample solution is to visit Server Pages JSPs remotely from WebSphere 7 to JReport Clustered Server using customized dispatcher. See the below diagram for the structure:



You should be able to set up a similar service with any Java EE server by following the same procedure based on your preferred application server documentation.

This demo dispatcher dispatches requests from different sessions to different JReport Servers according to Round-Robin algorithm. The dispatcher has the Fail Over function, which will periodically check whether there is any unavailable server in the cluster. No request will be dispatched to the unavailable server until the server is checked to be available again.

In general, the solution can be categorized into the following major steps:

1. Set up the server cluster.
2. Generate a WAR file containing Server Pages RMI JSP and dispatcher for WebSphere.
3. Deploy the WAR file to WebSphere.
4. Configure a clustered JReport Server.

In the following sections, we will explain clearly the exact operations you are expected to make in each of the main steps.

Setting up the server cluster

In this example we will set up two into the cluster using the Round-Robin algorithm.

192.168.0.1

192.168.0.2

Refer to Setting up and configuring a JReport Server Cluster for the specific steps of setting up JReport Server in a cluster.

Generating a WAR file containing Server Pages RMI JSP and dispatcher for WebSphere

1. Build a JReport Server WAR file as defined by makewar.xml for remote integration. The generated WAR file is saved to the default directory `<install_root>\bin\distribute`.

   ```
   makewar.bat buildRemoteWar -Djrs.remote.host=192.168.0.1 -Djrs.remote.rmiport=1129 -Djrs.rmi.auth_file=C:\JReport\Server\bin\rmi.auth
   ```

2. Compile the dispatcher DemoRemoteDispatcher.java stored in `<server_install_root>\help\samples\APICluster` with `<server_install_root>\lib\JRESServlets.jar` and `<server_install_root>\lib\servlet.jar`. The class file DemoRemoteDispatcher.class and some other class files will be generated.
3. In the WAR file, drag the class files generated in step 2 to the remote.war\WEB-INF\classes\demodispath folder, assuming that this folder has already been created in the WAR file.

Deploying the WAR file to WebSphere

1. Start IBM WebSphere 7.
2. Open Administrative Console. You can open Administrative Console by using the Start Menu, or by using the URL: `http://hostname:9060/ibm/console`, where hostname is host name or IP address, and 9060 is the port number.
3. After successfully logging in, expand the **Applications** node, and then click **Install New Application**.
4. Click **Browse** to select your .war file. In the Context root field, type a context path such as /servlet/ ("/servlet" is also ok). Click **Next**.
5. Do not check any option in this page, and then click **Next**.
6. Type **RMI_Server Pages** in the Application name field. DO NOT check the **Precompile JavaServer Pages files** option. Click **Next**.
7. Do not make any changes in the next two pages.
8. Click **Finish** in the Summary page. The installing process may take several minutes, wait until the process is completed.
9. After the installation process is completed, click **Save directly to the master configuration**. Then in the Save directly to the master configuration dialog, click **Save**.
10. This step is to configure the dispatcher and cluster server. Go to WebSphere Admin Control to add some properties for this dispatcher. Expand **Servers**, go through **Server Types** > **WebSphere application servers** > **server1** > **Process definition** (in the Server Infrastructure table > Java and Process Management) > **Java Virtual Machine** > **Custom properties** (in the Additional Properties table).

    Click the **New** button to add properties for our demo dispatcher com.jinfonet.dispatcher.configFile and jrs.remote.dispatcher.

    Figure 1: Add property for com.jinfonet.dispatcher.configFile

    Figure 2: Add property for jrs.remote.dispatcher

    Note that the dispatcher DemoRemoteDispatcher.java will read the clustered server information in the hostport.properties file like below:

    ```
    rmiserver=192.168.0.1:1129
    rmiserver=192.168.0.2:1130
    ...
    ```

11. If you have set up JReport Server in a cluster, you can append their host and port information to the above text file.
12. Click the **Save** link in the Messages table to save the changes, and then restart WebSphere 7.

Configuring the JReport Server Cluster

1. Make sure JReport Server has been started once in order that the server.properties file is generated.
2. Change server.properties file in `<server_install_root>\bin` as follows:

   ```
   server.rmiserver.enable=true
   server.rmiadminservice.enable=true
   ```

Then you can start JReport Server and access your Server Pages with a URL such as:

`http://hostname:9080/remote`

JReport Security System

In JReport Server, there are two types of security mechanisms. One is to deal with report resources, and the other is to deal with web pages.

This chapter discusses these two mechanisms respectively:

- Security for report resources
- Security for accessing web pages

Security for report resources

In JReport Server, there are two types of security mechanisms for managing access and visibility of report resources.

One is based on setting attributes of a report to control access and to control visibility of different subsets of data based on user identity. This control is set during the design step, carried forward when published, and handled at runtime without any special administrative action.

The other is based on administrative action in JReport Server to define the access rules for report resources in the runtime environment. This involves registering JReport Server users, and setting permissions on published Report resources.

This section discusses these two mechanisms respectively.

- Report security system
- Server security system

Report security system

This section demonstrates the control of access to reports and to different subsets of data by means of defining report security. In general, report security can be classified into three groups: cached report bursting, record level and column level. After applying report security, an end user will only see what he or she is allowed to see.

The following topics explain the report security at each level in detail:

- Cached report bursting
- Record-level security and column-level security
- Member-level security

Cached report bursting

Security in a report is a kind of privileged control. JReport supports cached report bursting which creates a security mechanism for controlling access to the report. By defining which groups of data are available to which users, groups, or roles, report results are created for each user, role and group. When a user accesses the report result, JReport checks the user, group and role of the user and merges the groups of data in the report the user is authorized to see and displays it to the user. Cached report bursting is implemented with these security properties on the group panel: Cascade, Grant, Groups, and Roles. The feature enables different users to view different data groups according to their access privileges. It also applies to nested groups.

This section focuses on how to view and schedule a report that has cached report bursting with JReport Server.

- Viewing a report with cached report bursting
- Scheduling a report with cached report bursting
- Example: e-mailing billing reports

Viewing a report with cached report bursting

Since the control of report access is not possible without a user ID, the significance of this function is only apparent after reports have been published to JReport Server and users access it using their JReport User ID (security identifier).

When a client views a report with cached report bursting in JReport Server, the corresponding groups will be displayed according to the security identifier. You can also advanced run reports with cached report bursting in different formats, including Page Report, HTML, PDF, TEXT, Excel, PS, XML and Rich Text Format (this feature does not support the RST and Applet formats).

To view a report with cached report bursting in JReport Server, the report must first be published to the server from JReport Designer. For example, if in JReport Designer, the security for a report that is grouped by the Customer_Region field has been set as follows:

- The user ID admin has the privilege to view the CA and MN groups of the report.
- The user ID jennifer has the privilege to view the BC group of the report.

Then,

1. Access the JReport Console page via a web browser with the user ID admin.
2. Browse to the report that you are going to view.
3. Click the report name, and you will then be able to view the CA and MN groups of the report.
4. If you log onto JReport Server with the user ID jennifer, you will then only be able to view the BC group.

Note: When designing the report in JReport Designer, if the Cascade property is set to be false, the specified group will only display its group header and footer.

Scheduling a report with cached report bursting

You can schedule a report with cached report bursting as a normal report. However, there are some differences between the formats in which the report is to be published.

Scheduling to HTML/Page Report Result to version

When you schedule a task to publish a report with cached report bursting to the HTML and/or Page Report Result formats to the versioning system, the scheduled result depends on the mode which is controlled by the property server.enable.cachedreportbursting in server.properties in `<install_root>\bin`:

- When `server.enable.cachedreportbursting=true` which is the default, JReport will create a report result with the report data for all possible users. This allows the report to run with a single query to the DBMS to create the report for all users in one pass. It is similar to report bursting, however, report bursting does not support the Page Report Result format and it makes a separate physical report result for each user and the administrator needs to manually restrict access to the result. When a user views the report result using cached report bursting, JReport Server will use the security identifier of the user to restrict access to the data in the report to the specific groups the security identifier is allowed to view. The end result is the same as the bursting report in that the user sees only his data, the advantage to the administrator is there is only one version result to manage. Users can perform interactive actions on the scheduled page report result as on other page report result, and the formulas, summaries and other similar data will be recalculated based on the privileged data.

  Note: If the report is cached report bursting and RLS/CLS mixed, then when other users other than the user who did the scheduling view the scheduled HTML result, a blank page is displayed.

- When `server.enable.cachedreportbursting=false`, the scheduled result only contains the data that the user who did the scheduling is allowed to see. This is primarily for compatibility with pre-8.2 versions of JReport. In this case, "to page report result" is not supported.

Scheduling to e-mail

When you schedule a report with cached report bursting to publish it to e-mail, there is a slight difference. JReport Server supports a multiple mail feature which enables sending the data result directly to each user who is authorized to view the report.

Assuming that the catalog and the report have been published to JReport Server, and two users admin and jennifer both have the permission to view the report. The following procedure shows how to schedule a task on a report with cached report bursting to be published to e-mail.

1. Access the JReport Console page with user ID admin or jennifer.
2. Browse to the row that the report is in, put the mouse pointer over the report row and click the **Schedule** button on the floating toolbar.
3. In the General tab, select the report tab with cached report bursting.
4. In the Publish tab, switch to the **To E-mail** sub tab and then check **This report has Cached Report Bursting. E-mail the report to each specified user**.
5. Type the subject and select the result format, then click **Finish**.

JReport Server will get the e-mail addresses from the user accounts, and then send the report result to admin and jennifer, with the contents in accord with their access right to the report.

Note: Before publishing to e-mail, make sure you have input the e-mail addresses of the users when configuring JReport Server. To do this:

1. Access the JReport Administration page, click **Security** on the system toolbar, and then select **User** from the drop-down menu.
2. In the User panel, choose the user name that you want to edit in the User ID column, and then click it. You can then type in the e-mail address of the user.

Scheduling to other formats

When scheduling a report with cached report bursting to other formats, the scheduled result only contains the report data that the user who does the schedule is allowed to see. All users will see the same data as the scheduling person.

Example: E-mailing billing reports

Sometimes, sending pertinent report data to corresponding mail recipients is required. JReport Server allows you to send a scheduled report result to the e-mail addresses accordingly, based on the cached report bursting settings of the report and the user information stored in JReport Server (for example, e-mail address information). Therefore, each recipient will only be able to view certain parts of the report data. However, using e-mail information of server users is not reliable, since a user ID that is appropriate for the report doesn't always exist in JReport Server. In cases like this, the mails will not be sent successfully. The best way to resolve this is to use an external e-mail information source by implementing the UserMailList and UserMailListFactory API that JReport provides.

JReport provides two interfaces for you to retrieve user and e-mail information from a customized source:

- **jet.server.api.UserMailListFactory**
  Used to get the UserMailList instance implemented by the user.
- **jet.server.api.UserMailList**
  Used to get e-mail information from a customized source.

You can implement multiple classes of interface UserMailList. Each of them may refer to a particular report. By using the getInstance() method in the jet.server.api.UserMailListFactory interface, you can get one implementation of the UserMailList. For more information on these two interfaces, see JReport Server API Documentation.

The following is a simple example:

1. Here, there is a Customers table with customer names and their e-mail addresses, as illustrated below. You can design a report using this table and others. Then apply cached report bursting for this report, so that later in the server side, you can schedule the report and send pertinent data to different recipients saved in this Customers table.

Customer Name	Customer Email
Absolute Java	joseph@abjava.com
American Coffee, Inc.	samuel@amcoffee.com
Automatic Java Machines	joan@autojava.com
Bob's Coffee	robert@bobcoffee.com
Brazilean's Best	hae@BrazilCoffee.com
Brew and Serve	kimberly@brewcoffee.com

2. In JReport Designer, design a report and set cached report bursting for it. In this case, group the report data by Customer Name, and then grant a formula FPageLevel to it. The content of the formula FPageLevel is as follows:

   ```
   @"Customer Name";
   ```

   This means that only the records of the specified group will be shown at runtime when you enter with different IDs -- Customer Name, in this example.

3. Then in the server side, implement the two interfaces to import the e-mail lists saved in the Customers table. Specifically, implement the jet.server.api.UserMailListFactory interface. The getInstance() method should be implemented in this interface to get an instance of jet.server.api.UserMailList. Take the following implementation as a reference, where the implementing class name of the UserMailList interface is formatted as "UserMailList_" + report + "_Impl", such as "UserMailList_InvoiceReport_cls_Impl".

```
import jet.server.api.*;
import jet.cs.util.*;

public class DemoUserMailListFactoryImpl implements UserMailListFactory {
    public UserMailList getInstance(ServerInfo serverInfo) {
        if (serverInfo == null) {
            return null;
        }
        String rpt = null;
        try {
            rpt = serverInfo.getTaskProperties().getProperty(APIConst.TAG_REPORT);
        } catch (RptServerException e) {
            e.printStackTrace();
            return null;
        }
        rpt = rpt.substring(rpt.lastIndexOf("/") + 1);
        rpt = rpt.replace('.', '_');
        rpt = rpt.replace(' ', '_');
        String clsName = "UserMailList_" + rpt + "_Impl";
        try {
            UserMailList mailList = (UserMailList)Class.forName(clsName).newInstance();
            return mailList;
        } catch (InstantiationException e1) {
            e1.printStackTrace();
        } catch (IllegalAccessException e1) {
            e1.printStackTrace();
        } catch (ClassNotFoundException e1) {
            e1.printStackTrace();
        }
        return null;
    }
}
```

4. Then implement the jet.server.api.UserMailList interface, which gets the user and e-mail information from the customized data source. The following implementation gets the e-mail list from the Customers table, which contains Customer Name and Customer E-mail columns.

```
import jet.server.api.*;
import java.util.*;
import java.sql.*;

public class UserMailList_InvoiceReport_cls_Impl implements UserMailList {
    public static Hashtable userEmails = new Hashtable();
    private String curRealmName = "defaultRealm";

    public UserMailList_InvoiceReport_cls_Impl() {
        loadData();
    }

    private void loadData () {
        try {
            String jdbcDriver = "sun.jdbc.odbc.JdbcOdbcDriver";
            DriverManager.registerDriver((Driver) Class.forName(jdbcDriver)
                    .newInstance());
            Connection conn = DriverManager
                    .getConnection("jdbc:odbc:jinfonet4");
            Statement stmt = conn.createStatement();
            ResultSet rs = stmt.executeQuery("select * from Customers");
            String userName = null;
            String userEmail = null;
            while(rs.next()) {
                userName = rs.getString("Customer Name");
                userEmail = rs.getString("Customer Email");
                if (userEmail != null) {
                    userEmails.put(userName, userEmail);
                }
            }
        } catch(Exception e) {
            e.printStackTrace();
        }
    }

    public java.util.Enumeration getAllMailAddresses(String realmName) {
        if (realmName.equals(curRealmName)) {
            return userEmails.elements();
        } else {
            return null;
        }
    }

    public java.util.Enumeration getGroupMailAddresses (String realmName, String groupName) {
        if (realmName.equals(curRealmName) && userEmails.containsKey(groupName)) {
            Vector groupEmails = new Vector();
            groupEmails.addElement(userEmails.get(groupName));
            return groupEmails.elements() ;
        } else {
            return null;
        }
    }

    public java.util.Enumeration getRoleMailAddresses (String realmName, String roleName) {
        if (realmName.equals(curRealmName) && userEmails.containsKey(roleName)) {
            Vector roleEmails = new Vector();
            roleEmails.addElement(userEmails.get(roleName));
            return roleEmails.elements() ;
        } else {
            return null;
        }
    }

    public java.lang.String getMailAddress(String realmName, String userName) {
        if (realmName.equals(curRealmName) && userEmails.containsKey(userName)) {
            return (String)userEmails.get(userName);
        } else {
            return null;
        }
    }
}
```

5. Register the above classes to JReport Server before the server is started.
   a. Add the parameter -Dcom.jinfonet.mailListFactory=UserMailListFactoryImplName to the command line/batch file that starts JReport Server, where UserMailListFactoryImplName indicates the implementation of the jet.server.api.UserMailListFactory interface.

      In this case, the parameter should be -Dcom.jinfonet.mailListFactory=DemoUserMailListFactoryImpl.

   b. Add the path of the implementation classes to the class path of the command line/batch file.
6. Start JReport Server and then publish the report and catalog.
7. Schedule the report, publish to e-mail, check the option **This report has Cached Report Bursting. E-mail the report to each specified user**. Provide the necessary information and then submit the schedule.

The report will be processed and sent to the corresponding recipients with pertinent report data.

Record-level security and column-level security

The record-level security (RLS) and column-level security (CLS) of JReport Designer allow you to control user access to different subsets of data and ensure that people only see what they are supposed to see: record-level security allows you to define which records are to be revealed to any given user, while column-level security allows you to define which report column is revealed to any given user. This enables you to provide different users with accordingly different, but appropriate contents. No matter to whom you need to provide information, a plant manager or thousands of customers, JReport Designer allows you to control access to information according to your requirements.

JReport Designer have two types of the security policies, one is a security policy based on a data source connection (connection-scope security policy), and the other is a security policy based on a single report (report-scope security policy).

- **Connection-scope security**
  You can build connection-scope security policies: where each security policy refers to a data source connection in the catalog. If you want to implement the same security policy in a group of reports, you can simply apply an existing security policy to the report, without having to repeatedly build security information for each report. Both RLS and CLS can be connection-scope security.

- **Report-scope security**
  Additionally, record-level security can be of report scope, based on the security information file. That is, you can use the security information file to set the security policies for a report. Report-scope security policy doesn't support column-level security.

Record-level security can be applied simultaneously to both connection and report scopes. However, if a report-scope security policy has already been applied to a report, it will override a connection-scope security policy applied to the report. That is, report-scope security policies have a higher priority than connection-scope security policies.

Running reports with record-level/column-level security

After setting up the security policy for a report in JReport Designer, you can then publish it to JReport Server as normal. Then, when you log onto the server as different users, you will find that the security settings are applied to the report. That is, different users will only see the data they are supposed to see.

Note: The report designer defined users and roles may not be recognized by JReport Server. If your security policies contain such users/roles, create these users and roles respectively in JReport Server, and then in JReport Designer, synchronize the security information with JReport Server using the Merge option.

Member-level security

Member-level security (MLS) enables report designers to limit user access to the specific members of groups in business views. By defining which members of a group are available to which users, groups, or roles, report results are created for each user, role and group. When a user accesses a report, JReport checks the user, group and role of the user and merges the data in the report the user is authorized to see and displays it to the user.

The relationship between a principal (user, role or group) and the members of a group are classified into three groups:

- **Allowed Set**
  The members of a group visible to a principal, which are directly specified to the principal, excluding those inherited from the parent roles or groups.
- **Denied Set**
  The members of a group invisible to a principal, which are directly specified to the principal, excluding those inherited from the parent roles or groups.
- **Unspecified members**
  The members of a group that are not specified in a user's allowed/denied set or in the inherited allowed/denied set from the user's parent roles/groups. Report designer can determine whether the unspecified members are visible to a user.

A principal can have its own allowed/denied set and inherit the allowed/denied sets from its parent roles or groups. The parent allowed/denied sets will be calculated first and it is a recursive process.

Server security system

All of the report resources that are published to JReport Server are protected by the JReport Server security system. This system is based on maintaining a registered set of users and setting permissions on each resource for each user.

The User data model defines users, as well as their membership in groups and roles. By having groups and roles, the system can apply permissions for a set of users with one transaction.

Resources are objects in resource trees or version tables, such as folders, resources and archive versions. The permission settings control capability for a given user on the resource - if the resource is visibile, if it can be read, written, deleted, executed, scheduled, and have its status updated.

JReport Server supports two ways to apply permissions to the the set of users. One is the default system of setting permissions for users, groups and roles. The other is role-based definition, in which permissions are defined on roles only, and users and groups are mapped to roles.

When JReport Server is performing runtime security checking for a given user, it will respect the permissions settings and follow the access control rules when processing the service requests.

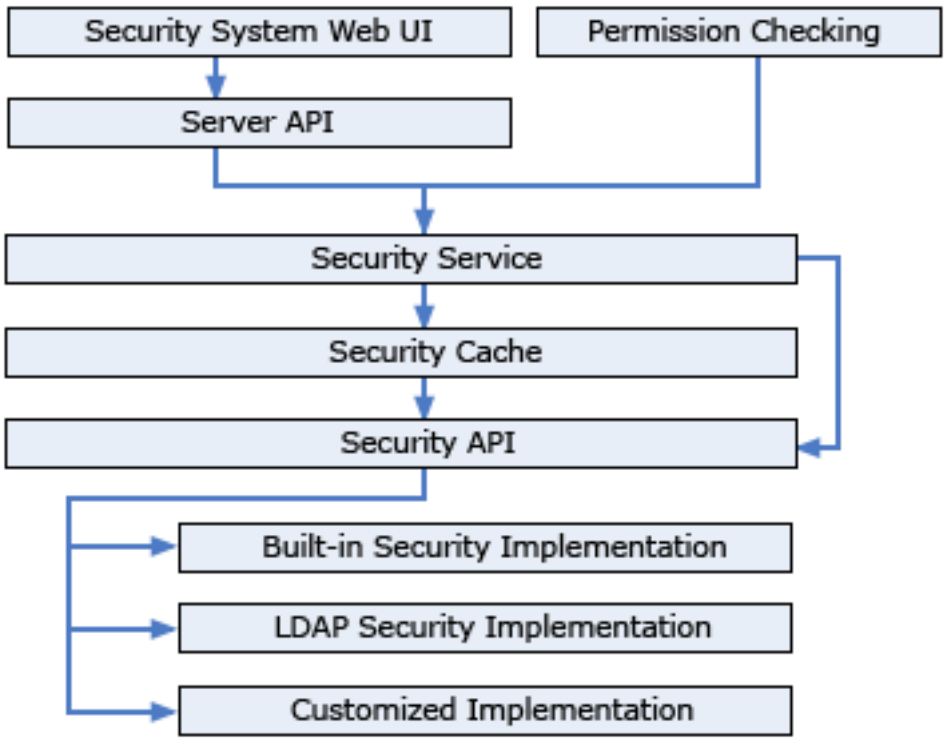
The JReport Server runtime security checking system is implemented based on a standard set of Security Service method calls. The default implementation provided in JReport Server is based on the data set of users and resources that is stored in JReport Server.

For integration with an existing application that already has a system for managing users and permissions, JReport Server defines the Security Service as a Java interface and allows for an application developer to supply a customized version to replace the default implementation. This allows an existing application to provide a custom version of the Security Service that supports JReport Server runtime security checking based on user data and permissions that are stored outside of JReport Server. In this configuration, the JReport Server admin section for managing users and permissions is not used.

JReport Server also can integrate with an existing application that uses an LDAP system for managing user and group information. JReport Server can be configured to interact with the LDAP system so that edit of information about users and groups can be done only in the LDAP system. Informaton about permissions for resources is not part of the LDAP data model. That information continues to be maintained by JReport Server.

Accessing user and permission data by database lookup on each service request may result in many time-consuming IO operations. As a result, the performance of the server security system may be lowered. In order to promote performance, a cache system exists just above the Security Service. The cache system is used to store security objects including users, groups, roles and access control lists obtained from making calls to the Security Service. This cached data will be used when the same information is needed later.

The following is a diagram of the JReport Server security system structure:



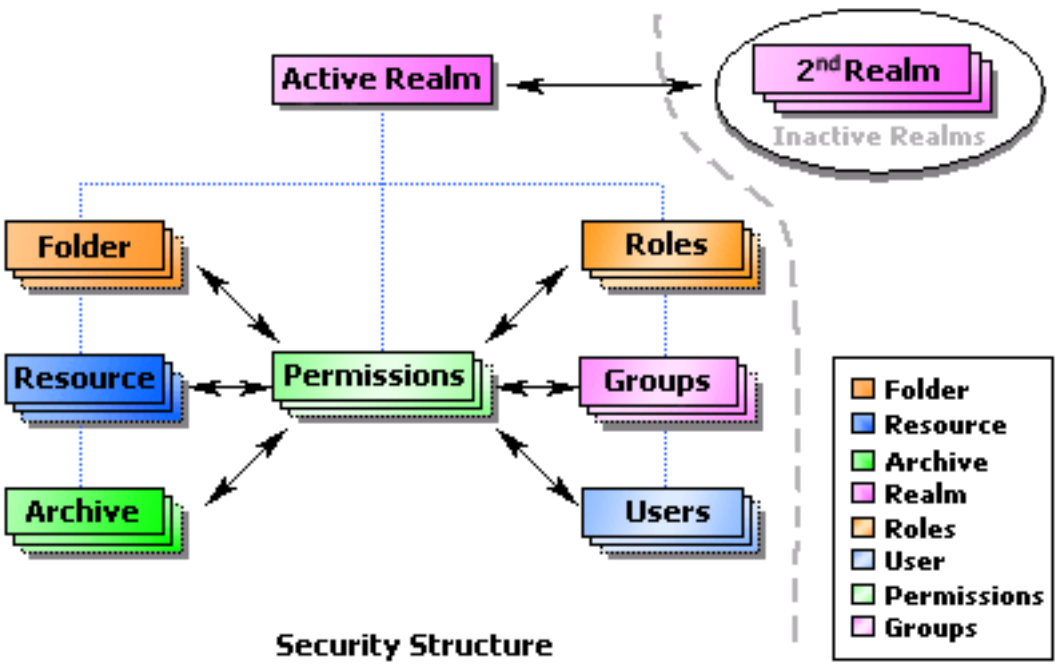
Go through the following topics for details about the server security system:

- Security system data model
- Role based security
- Security cache system
- Customized implementation of the Security API
- Using an LDAP server's security system

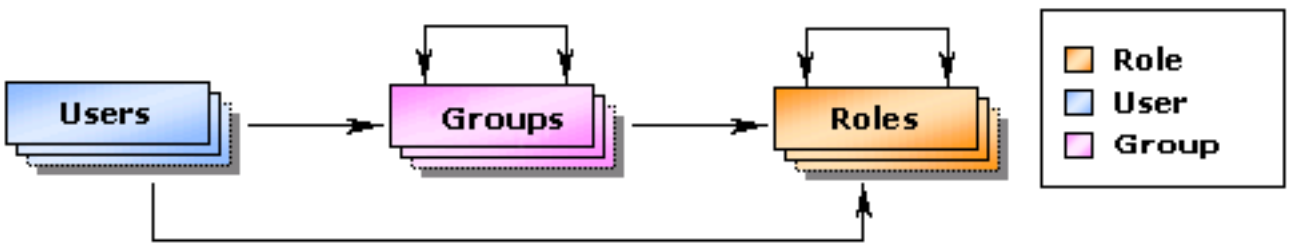
Security system data model

JReport Server provides a database to hold registered users and permissions on report resources. This database will be used by the JReport Server Security Service to control access at runtime.

This diagram illustrates the data model used by the Security Service in JReport Server:



This shows the inherited relationship among User, Group and Role.



User data model

JReport Server defines these entities in the user model:

- **Realm**
  Realm is an abstract security concept, which hosts the resources and authentication entities on JReport Server. There can be more than one realm on the server and each realm is independent from others. The resources and authentication entities that reside in different realms are different.

  At runtime, only one realm can be active and only the users and resources in the active realm are accessible. A realm is identified by a unique name, which can contain any characters other than forward slash (/) and backward slash (\).

  The authentication entities consist of user accounts, group accounts and role accounts.

- **User**
  The user is the primary element of the database. It is a unique name that identifies a particular user of JReport Server.

  The JReport Server web pages are not accessible until a servlet session is established based on verification and login of a registered JReport Server user. All activity done during the web session will use that user identity when the Security Service is considering access privileges to report resources.

  JReport Server comes with two built-in users, admin and guest. The built-in user accounts cannot be deleted. The Admin user account can neither be deleted nor disabled.

- **Group**
  The principal group, which represents an organization of user accounts, is available for managing users. Users or groups can be added into a group as its child members, and therefore inherit the resource and folder permissions from the group.

- **Role**
  Users must have certain user rights and permissions in order to perform tasks on resources. Roles, which represent an aggregate of permissions, help you to efficiently assign the appropriate user rights and permissions to users. Assigning roles to users gives the users all of the user rights and permissions of the roles to perform their jobs with. A role can also be assigned to other groups or roles, and thus groups or roles can inherit the permissions of other roles. JReport Server comes with two built-in role accounts, administrators and everyone. The built-in role accounts cannot be deleted. The administrator role account can neither be deleted nor disabled.

Access control data model

Access control to report resources is based on this data model:

- **Permission**
  Permissions, associated with resources and folders, are the rules that are granted to users to control their access to the resources and folders.

- **Privilege**
  Privilege is a mode for managing permissions. It can be used to manage different access permissions unrelated with nodes. JReport Server offers these types of privileges for users: Publish and Advanced Properties. Users that are granted the Publish privilege will be able to publish resources to JReport Server, while users that have the privilege of Advanced Properties are allowed to view advanced information of version properties such as catalog connections and report related resources.

- **Alias**
  JReport Server organizes file and directories into a Resource Tree. Aliases are used to provide different "views" of a tree for different users to enter the Resource Tree. For example, you may set an alias resource tree (based on the resource tree) for Tanya, so that she can only see the market resource node and thus can directly walk into the report file she is interested in. In summary, an alias is a combination of users and resource nodes.

To manage JReport Server's built-in security, you must be a member of the administrator role in order to access the JReport Administration page.

Related topics:

- Managing security

Role based security

In addition to the security system based on users, groups and roles, JReport Server also supports a role based security system in which permissions are defined on roles only, and users and groups are mapped to roles.

To switch to the role based security system, you can use either of the following two methods:

- Check the **Role Based Authorization** option on the JReport Administration page > Configuration > Advanced panel.

  By default, the option is unchecked and the security mechanism of setting permissions for users, groups and roles is applied. If the option is checked, the Permission Setting UI Displays option will be hidden automatically, since it is used to control UI display and is of no use in the role based security environment.

- In the server.properties file, set server.rolebased.authorization to **true**.

  In this case, the following three properties will not take effect since they control UI display of setting user/group/role permissions:

  server.ui.set_permissions.group

  server.ui.set_permissions.role

  server.ui.set_permissions.user

If the role based security system is used, in both the JReport Administration page (port 8889 by default) and Console page (port 8888 by default), when you set permissions of a resource node, there are only roles displayed. Role only based security is similar to Java EE security where the developer assigns roles and during deployment users and groups can be mapped to the roles so if you are already using Java EE security, this would be the best method.

Security cache system

The security cache system temporarily stores security objects such as users, roles, groups and ACLs. ACL, short for Access Control List, is the core object of the security authorization system, and is in charge of storing and checking principal permissions. When JReport Server requires information from the security system, it can fetch it from the cache for better performance.

The cache system caches not only security objects for the built-in security system, but also those implemented by the Security API from the external security system. It caches security information in the security data. If the security service needs security information, it will fetch it from the security data. However, if the security data cannot find the information, it will request it from the Security API, and then cache it in the cache system. When the security information is modified in the security system, the Security API is invoked directly in order to modify the security data.

Note: There is a special interface SecurityListener in the cache system, through which the cache is noted to update the cached information. It is recommended that you invoke it when you access the external security system, so as to synchronize security data between the cache system and the external security system.

The following focuses on the configuration and synchronization of the security cache system.

Configuring the security cache system

The security cache system enables you to define the maximum number of users, roles, groups and ACL objects that can be cached. There are three ways in which you can customize the security cache system as explained below:

Configuring by editing the server.properties file

Edit the following four properties:

- **server.security.user.cache.size**
  This should be an integer value. Its value indicates the maximum number of user objects that the security cache can store. The default value is 1000.
- **server.security.role.cache.size**
  This should be an integer value. Its value indicates the maximum number of role objects that the security cache can store. The default value is 50.
- **server.security.group.cache.size**
  This should be an integer value. Its value indicates the maximum number of group objects that the security cache can store. The default value is 50.
- **server.security.protection.cache.size**
  This should be an integer value. Its value indicates the maximum number of ACL objects that the security cache can store. The default value is 100.

For instance,

- If server.security.user.cache.size=1000, the cache can then store at most 1000 user objects.
- If server.security.role.cache.size=100, the cache can then store 100 role objects.
- If server.security.group.cache.size=100, the cache can then store 100 group objects.
- If server.security.protection.cache.size=100, the cache can then store 100 ACL objects.

Configuring from the JReport Administration page

You must be a member of the administrator role in order to access the JReport Administration page.

1. Log onto the JReport Administration page, click **Cache** on the system toolbar, and then select **Security Cache** from the drop-down menu.
2. In the Security Cache panel, four options are provided for specifying the cache size:
   - **User Cache Size**
     The maximum number of user objects that the security cache can store. Should be an integer value.
   - **Role Cache Size**
     The maximum number of role objects that the security cache can store. Should be an integer value.
   - **Group Cache Size**
     The maximum number of group objects that the security cache can store. Should be an integer value.
   - **Protection Cache Size**
     The maximum number of ACL objects that the security cache can store. Should be an integer value.
3. When done, click **Save** to apply the settings.

Configuring using the API method

Invoke the following methods in the API class jet.server.api.admin.cfg.ConfigurationAdvanced:

```
/**
 * Set the security user cache's size
 * Setting the size of the cache to zero or negative means closing the security user cache.
 * @param size
 */
public void setSecurityUserCacheSize(int size);

/**
 * Get the size of the security user cache
 * @return the size of the security user cache
 */
public int getSecurityUserCacheSize();

/**
 * Set the size of the security role cache
 * Setting the size of the cache to zero or negative means closing the security role cache
 * @param size
 */
public void setSecurityRoleCacheSize(int size);

/**
 * Get the size of the security role cache
 * @return the size of the security role cache
 */
public int getSecurityRoleCacheSize();

/**
 * Set the size of the security group cache
 * Setting the size of the cache to zero or negative means closing the security group cache
 * @param size
 */
public void setSecurityGroupCacheSize(int size);

/**
 * Get the size of the security group cache
 * @return the size of the security group cache
 */
public int getSecurityGroupCacheSize();

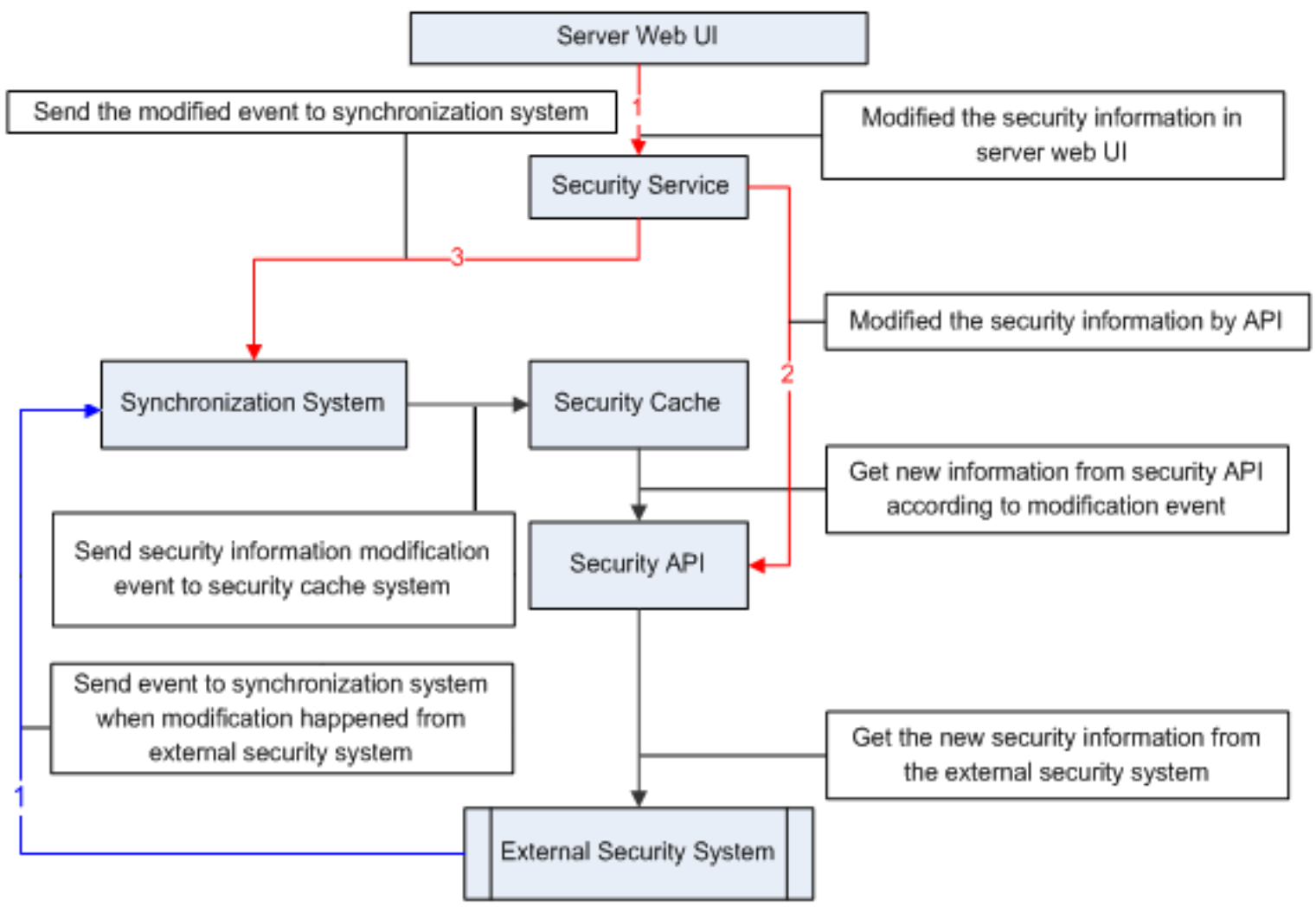
/**
 * Set the size of the security protection cache
 * Setting the size of the cache to zero or negative means closing the security protection cache
 * @param size<
 */
public void setSecurityProtectionCacheSize(int size);

/**
 * Get the size of the security protection cache
 * @return the size of the security protection cache
 */
public int getSecurityProectionCacheSize();
```

Synchronizing the security cache system

A synchronization system has been provided for synchronizing JReport Server's security system with your external security systems. When the security cache system receives a security information modification event, it will then fetch the security information from the API and update the cached information.

The following is a diagram of the synchronization system mechanism:



There are two ways to invoke the synchronization system. The first is to modify the security information on our Server web UI (red line), and the second is to modify the external security system (blue line).

Customized implementation of the Security API

JReport Server provides a set of Security APIs which you can implement in order to build your preferred security system. This section discusses the basic rules and implementation of the Security API.

Partial Implementation of the Security API

The Security API can meet various requirements for seamlessly integrating the JReport security system into an existing external security system.

By implementing all interfaces, you provide full functions to the JReport Server security system. However, if you do not want to provide a complete security system to your JReport Server, but only to use part of the functions available, you can achieve this by implementing the interfaces that meet your requirements.

Here are some examples:

Example 1

Requirements: You only want to customize authentication and authorization.

The following interfaces are required to be implemented:

- AuthenticationProvider
- AuthorizationProvider

Example 2

Requirements: You want to customize authentication and users, while the JReport Server maintains other functions of the security system.

The following interfaces are required to be implemented:

- AuthenticationProvider
- UserProvider

Example 3

Requirements: For CLS/RLS scenario, you must provide user/role information.

The following interfaces are required to be implemented:

- AuthenticationProvider
- UserProvider
- RoleProvider
- RoleUserRelationProvider

Implementing the Security API using an .xml file

You can specify a customized implementation of the Security API in a .xml file. The JReport Server loads classes according to this file.

The file is customizedAPI.xml in `<install_root>\bin`. Specify the content in the .xml file as follows:

```
<?xml version="1.0" encoding="UTF-8"?>

<jreport-customized-api>
    <security>
        <authentication-provider>
            com.customer.security.AuthenticationProviderImpl
        </authentication-provider>
        <authorization-provider>
            com.customer.security.AuthorizationProviderImpl
        </authorization-provider>
        <user>
            <provider>
                com.customer.security.user.UserProviderImpl
            </provider>
            <permission-provider>
                com.customer.security.user.UserPermissionProviderImpl
            </permission-provider>
            <privilege-provider>
                com.customer.security.user.UserPrivilegeProviderImpl
            </privilege-provider>
        </user>
        <group>
            <provider>
                com.customer.security.group.GroupProviderImpl
            </provider>
            <permission-provider>
                com.customer.security.group.GroupPermissionProviderImpl
            </permission-provider>
            <privilege-provider>
                com.customer.security.group.GroupPrivilegeProviderImpl
            </privilege-provider>
        </group>
        <role>
            <provider>
                com.customer.security.role.RoleProviderImpl
            </provider>
            <permission-provider>
                com.customer.security.role.RolePermissionProviderImpl
            </permission-provider>
            <privilege-provider>
                com.customer.security.role.RolePrivilegeProviderImpl
            </privilege-provider>
        </role>
        <relation>
            <role-group>
                com.customer.security.relation.RoleGroupRelationProviderImpl
            </role-group>
            <role-user>
                com.customer.security.relation.RoleUserRelationProviderImpl
            </role-user>
            <group-user>
                com.customer.security.relation.GroupUserRelationProviderImpl
            </group-user>
        </relation>
    </security>
</jreport-customized-api>
```

Rules for applying customized Security API

The following are rules for applying customized Security API:

- AuthenticationProvider is a core interface and must be implemented. It is first loaded when JReport Server loads a customized Security API.
- There are dependency relationships among some interfaces:

  Interfaces PermissionProvider and PrivilegeProvider depend on the corresponding principal interface provider. For example, the prerequisite for applying a customized UserPermissionProvider is that a customized UserProvider has been applied.

  Interface RelationProvider depends on both the corresponding principal interfaces. For example, the prerequisite for applying a customized GroupUserRelationProvider is that a customized UserProvider and GroupProvider have been applied.
- If you partly implement the Security API, the JReport Server will automatically provide implementation of some missed but required interfaces, in order to build an integrated security system. Below are the rules:
  - If customized implementations of Providers have not been applied, the JReport Server will apply built-in implementations of these Providers, including: AuthorizationProvider, UserProvider, GroupProvider, RoleProvider, GroupUserRelationProvider, RoleGroupRelationProvider, and RoleUserRelationProvider.
  - If you have applied a PermissionProvider or PrivilegeProvider, but not applied an AuthorizationProvider, the JReport Server will apply the built-in implementation of the AuthorizationProvider.
- If you have applied an AuthorizationProvider, but not applied a PermissionProvider or PrivilegeProvider, for example, UserPermissionProvider or UserPrivilegeProvider, the JReport Server will not apply the built-in implementation of the PermissionProvider or PrivilegeProvider.

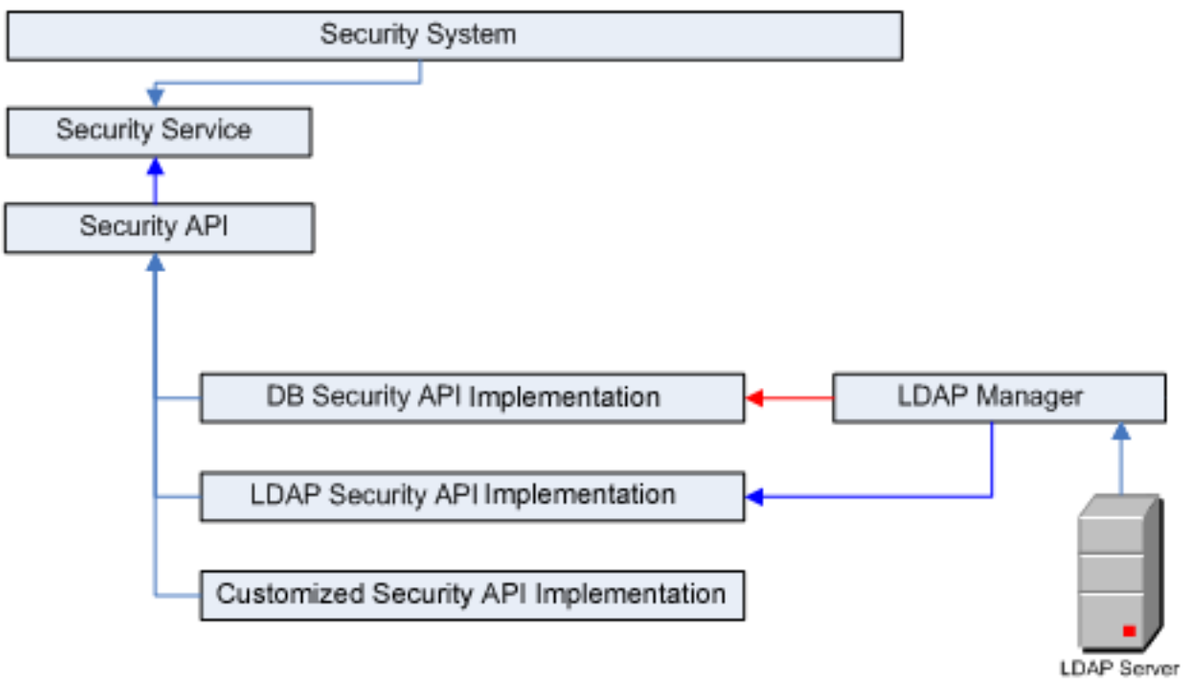
Notes:

- For details about the Security API, see the jet.server.api.custom.security package in the JReport Javadoc in `<install_root>\help\api`.
- Demo references available in `<install_root>\help\samples\APISecurity`:
  - **DemoAuthenticationProvider.java**
    Demo for implementation of the jet.server.api.custom.security.AuthenticationProvider interface.
  - **DemoAuthorizationProvider.java**
    Demo for implementation of the jet.server.api.custom.security.AuthorizationProvider interface.
- In V8, two new methods addSecurityListener() and isEnableEdit() are added into the API jet.server.api.custom.security.AuthenticationProvider. If you have upgraded your JReport Server from V7 to a higher version, and have applied customized implementation of AuthenticationProvider in V7, you need implement the new methods in the API.
- JReport Server provides the ability to use customized user authentication scheme by the implementation of the two interfaces jet.server.api.custom.security.AuthenticationProvider and jet.server.api.custom.security.AuthorizationProvider.

Using an LDAP server's security system

The server security system can run two modes in which you can use an LDAP server's security system. The first is importing mode. In this mode, if you want to use the LDAP feature, you will have to import the security information from an LDAP server. The second is non-importing mode. With this mode, JReport Server can access an LDAP server and obtain LDAP security information directly without having to import it.

Below is a diagram which illustrates these two working modes:



JReport Server can access an LDAP server using the LDAP implementation of the Security API (blue line), and import security information from an LDAP server into the built-in security system (red line).

To use an LDAP server's security system, you should first enable JReport Server to adapt to a directory server by configuring the settings in the Server tab (accessed by selecting **Configuration** > **LDAP** on the JReport Administration page), making sure that the Enable LDAP Support option has been checked.

Notes:

- Make sure that the Directory Manager DN is a user with prior LDAP Server permission, and who can retrieve other LDAP users.
- Make sure that the users and groups you want to query and import into JReport Server belong to the organization you typed into the Distinguished Name field.
- When the system built-in admin user logs in JReport Server, JReport always tries the system built-in password (admin by default) first. This is to provide a backdoor password for admin in case the LDAP server is down.

Pick a topic from the following to get more:

- Examples of LDAP server configuration
- SSL support in LDAP system
- Limitations of LDAP support
- Using LDAP server security information by importing
- Using LDAP server security information via the LDAP implementation of the Security API

Examples of LDAP server configuration

Six directory servers are currently supported. These are:

- Novell Directory Server
- Microsoft Site Server
- iPlanet Directory Server
- Active Directory Advanced Server
- Lotus Domino Server
- OpenLDAP Directory Server

If you need access to a different directory server, contact your JReport Sales Representative. New servers are frequently being added.

The following sections provide examples of settings which enable JReport Server to adapt to these directory servers. They are as follows:

- Example 1: Configuration for adapting to a Novell Directory Server
- Example 2: Configuration for adapting to a Microsoft Site Server
- Example 3: Configuration for adapting to an iPlanet Directory Server
- Example 4: Configuration for adapting to the Active Directory Advanced Server
- Example 5: Configuration for adapting to a Lotus Domino Server on NT
- Example 6: Configuration for adapting to an OpenLDAP Directory Server

Example 1: Configuration for adapting to a Novell Directory Server

If you want to get all users and groups from the orgunit organizational unit, you should follow the steps below to configure your Server tab on the JReport Administration page > Configuration > LDAP panel:

1. Select **Novell Directory Server** from the Select LDAP Server drop-down list, and then click **Load Settings**. The settings of the Novell Directory Server will then be loaded.
2. Check the **Enable LDAP Support** checkbox, and input the following information:
   - **LDAP URL:** ldap://IP address or host name of your Novell Directory Server (for example: `ldap://127.0.0.1`)
   - **LDAP Server Port:** 389
   - **Root Entry:** o=the name of the root (for example: o=myorg)
   - **Directory Manager DN:** cn=user name of the directory manager,o=context (for example: cn=admin,o=context)
   - **Password:** the password of the Directory Manager (for example: 1234)
   - **Encryption Type:** None
   - **Import LDAP Groups to:** Group
   - **User Schema**
     - Distinguished Name:ou=the name of the organization unit where you want to perform a search for users (for example: ou=orgunit)
     - Filter: (&(cn=the filter criteria that you want to set)(objectclass=person)) (for example: (&(cn=*)(objectclass=person)))
   - **Group Schema**
     - Distinguished Name: ou=the name of the organization unit that you want to perform a search for groups (for example: ou=orgunit)
     - Filter: (&(cn=the filter criteria that you want to set)(objectclass=groupofuniquenames)) (for example: (&(cn=*)(objectclass=groupofuniquenames)))
     - Admin Group: The name of the group you want to add to the Admin group (for example: develop)
3. You can test the connection settings by clicking the **Test Connection** button, get the query result of the users specified in the option Filter by clicking the **Query User** button, and get the query result of groups specified in the option Filter by clicking the **Query Group** button.
4. Click **Save** to save all settings.

Example 2: Configuration for adapting to a Microsoft Site Server

Follow the steps below to configure your Server tab on the JReport Administration page > Configuration > LDAP panel. You can get all users from the members organizational unit and all groups from the groups organizational unit.

1. Select **Microsoft Site Server** from the Select LDAP Server drop-down list, and then click **Load Settings**. The settings of the Microsoft Site Server will be loaded.
2. Make sure that the Enable LDAP Support checkbox is selected, and input the following information:
   - **LDAP URL:** ldap://IP address or host name of your Microsoft site Server (for example: `ldap://127.0.0.1`)
   - **LDAP Server Port:** 1003
   - **Root Entry:** o=test
   - **Directory Manager DN:** cn=administrator,ou=members,o=test
   - **Password:** test
   - **Encryption Type:** None
   - **Import LDAP Groups to**: Group
3. You can test connection settings by clicking the **Test Connection** button, get the query result of users specified in the option Filter by clicking the **Query User** button, and get the query result of groups specified in the option Filter by clicking the **Query Group** button.
4. Click **Save** to save all settings in this tab.

Example 3: Configuration for adapting to an iPlanet Directory Server

Follow the steps below to configure your Server tab on the JReport Administration page > Configuration > LDAP panel. You can get all users in the people organizational unit and all groups in the groups organizational unit.

1. Select **iPlanet Directory Server** from the Select LDAP Server drop-down list, and then click **Load Settings**. The settings of iPlanet Directory Server will then be loaded.
2. Make sure that the Enable LDAP Support checkbox is selected, and input the following information:
   - **LDAP URL:** ldap://IP address of your iPlanet Directory Server
   - **LDAP Server Port:** 389
   - **Root Entry:** dc=mailbj,dc=jinfonet,dc=com
   - **Directory Manager DN:** cn=directory manager
   - **Password:** jinfonet
   - **Encryption Type:** None
   - **Import LDAP Groups to**: Group
3. You can test the connection settings by clicking the **Test Connection** button, get the query result of users specified in the option Filter by clicking the **Query User** button, and get the query result of groups specified in the option Filter by clicking the **Group Query** button.
4. Click **Save** to save all the settings in this tab.

Example 4: Configuration for adapting to the Active Directory Advanced Server

Follow the steps below to configure your Server tab on the JReport Administration page > Configuration > LDAP panel. You can get all users and groups from the myorg organizational unit.

1. Select **Win2000 Active Directory** from the Select LDAP Server drop-down list, and then click **Load Settings**. The settings are then cleared and you can input your information.
2. Make sure that the Enable LDAP Support checkbox is selected, and input the following information:
   - **LDAP URL:** ldap://IP address of your Windows 2000 Advanced Server
   - **LDAP Server Port:** 389
   - **Root Entry:** DC=testad,DC=local
   - **Directory Manager DN:** CN=administrator,CN=Users,DC=testad,DC=local
   - **Password:** 1234
   - **Encryption Type:** None
   - **Import LDAP Groups to**: Group
   - **User Schema**
     - User Attribute Name: cn
     - User Common Name: userPrincipalName
     - User Password: userPassword
     - Distinguished Name: ou=myorg
     - Filter: (&(cn=*)(objectclass=person))
   - **Group Schema**
     - Group Common Name: cn
     - Group Member Type: member
     - Distinguished Name: ou=myorg
     - Filter: (&(cn=*)(objectclass=group))
3. You can test the connection settings by clicking the **Test Connection** button, get the query result of users specified in the option Filter by clicking the **Query User** button, and get the query result of groups specified in the option Filter by clicking the **Group Query** button.
4. Click **Save** to save all settings in this tab.

Example 5: Configuration for adapting to a Lotus Domino Server on NT

By following the steps below to configure your Server tab on the JReport Administration page > Configuration > LDAP panel, you can get all users and groups from the developer organization unit.

1. Select **Lotus Domino on NT** from the Select LDAP Server drop-down list, and then click **Load Settings**. The settings are then cleared and you can input your information.
2. Make sure that the Enable LDAP Support checkbox is selected, and input the following information:
   - **LDAP URL:** ldap://IP address of your Lotus Domino Server
   - **LDAP Server Port:** 389
   - **Root Entry:**
   - **Directory Manager DN:** cn=admin,o=jtotal
   - **Password:** 123456
   - **Encryption Type:** None
   - **Import LDAP Groups to**: Group
   - **User Schema**
     - User Attribute Name: uid
     - User Common Name: cn
     - User Password: userPassword
     - Distinguished Name: ou=developer, o=jtotal
     - Filter: (&(cn=*)(objectclass=person))
   - **Group Schema**
     - Group Common Name: cn
     - Group Member Type: member
     - Distinguished Name:
     - Filter: (&(cn=*)(objectclass=groupofnames))
3. You can test the connection settings by clicking the **Test Connection** button, get the query result of users specified in the option Filter by clicking the button **Query User**, and get the query result of groups specified in the option Filter by clicking the **Query Group** button.
4. Click **Save** to save all settings in this tab.

Example 6: Configuration for adapting to an OpenLDAP Directory Server

By following the steps below to configure your Server tab on the JReport Administration page > Configuration > LDAP panel, you can get all users and groups from the developer organization unit.

1. Select **OpenLDAP Directory Server** from the Select LDAP Server drop-down list, and then click **Load Settings**. The settings are then cleared and you can input your information.
2. Make sure that the Enable LDAP Support checkbox is selected, and input the following information:
   - **LDAP URL**: ldap://IP address of your OpenLDAP Directory Server (for example: `ldap://127.0.0.1`)
   - **LDAP Server Port**: 389
   - **Root Entry:** dc=openldap, dc=ldaptest
   - **Directory Manager DN**: cn=Manager,dc=openldap,dc=ldaptest
   - **Password**: 123456789
   - **Encryption Type**: None
   - **Import LDAP Groups to**: Group
   - **User Schema**
     - User Attribute Name: uid
     - User Common Name: cn
     - User Password: userPassword
     - Distinguished Name: ou=members
       Filter: (&(uid=*)(objectclass=person))
   - **Group Schema**
     - Group Common Name: cn
     - Group Member Type: uniqueMember
     - Distinguished Name: ou=groups
     - Filter: (&(cn=*)(objectclass=groupofuniquenames))
     - Admin Group:
3. You can test the connection settings by clicking the **Test Connection** button, get the query result of users specified in the option Filter by clicking the **Query User** button, and get the query result of groups specified in the option Filter by clicking the **Query Group** button.
4. Click **Save** to save all settings in this tab.

SSL support in LDAP system

JReport Server's LDAP system supports SSL when connecting to an LDAP server for obtaining security information.

Solving the wrong connection port type problem

JReport implements a method in the security system to solve the wrong connection port type problem.

The wrong connection port type problem in SSL protocol

If you use an SSL socket to connect to a server on a port that is not using SSL, or if you use a plain socket to connect to a server's SSL socket, your program will hang. This is a characteristic of the SSL protocol.

Method to avoid the wrong connection port type problem

Use a main thread to create a child thread for connecting to the LDAP server. The main thread can wait on the child thread for a period of time-- the socket timeout time (This time can be set by users). If the child thread creates an LDAP connection successfully, it will notify the main thread, and the program will continue to run. However, if the child thread hangs due to using the wrong port type, the main thread will only need to wait until the socket timeout time has been reached and can continue to run.

A Parameter setting in the method

There is an important parameter in this method: the socket timeout time. Since the connection time varies with the user's network environment, it is better to set it in LDAPProperties.xml: modify the element env-socketTime's value before the server is started. Its default value is 10, which means that the socket timeout time is 10 seconds. You can modify this value according to your network environment.

About SSL certificate store in JReport Server

Since JNDI uses the default SSL provider, the certificates will be checked by JSSE's default TrustManager: X509TrustManager. If the TrustManager does not accept them, JReport Server will store the SSL certificates into another key store file. This file is placed in `<install_root>\properties\LDAPKeyStore.keystore`. The password to access the file is *jinfonet*. You can also use `-D` parameters to specify another file and password. For instance, if you want to add the certificates into: `C:\certs\certs.keystore`, and use the password *test*, you should add the following parameters to JReport Server's startup file:

```
"-Djavax.net.ssl.trustStore= C:\certs\certs.keystore"
"-Djavax.net.ssl.trustStorePassword=test"
```

Note: The LDAP service provider uses JSSE for its SSL support. JSSE is available as part of Java 2 SDK, v1.4. As for earlier versions of the Java platform, you can turn to http://java.sun.com/products/jsse for information. To use JSSE on a platform earlier than Java 2 SDK, v1.4, first install JSSE, and then configure a JSSE provider either by updating the JAVA_HOME/lib/security/java.security file with the provider or by adding the provider programmatically. Here JAVA_HOME refers to the directory where the Java Runtime (JRE) software has been installed. Detailed steps can be found in the JSSE Reference Guide.

Limitations of LDAP support

In the LDAP page, sessions are used to remember passwords and the status of the remember password checkbox. Due to this, when the session exceeds the time limit, there are some limitations:

- If the Load Settings button is clicked, and the session has expired, the user must log onto JReport Server again. After that, the retained password will be lost.
- If the session expires due to idling, when logging back onto JReport Server again, the retained password will be lost.

Using LDAP server security information by importing

LDAP (Lightweight Directory Access Protocol) is a lightweight client-server protocol for accessing directory services. With LDAP support, JReport Server enables you to import users/groups from directory servers.

To import LDAP security server information, on the JReport Administration page, click **Configuration** > **LDAP** > **Import**. In the Import tab, you can use LDAP users and groups in JReport Server by importing them. In addition, in order to have the most current security information, you can synchronize the security information from your local server with that of the LDAP server. To do this, on the JReport Administration page, click **Configuration** > **LDAP** > **Synchronize**, where you can predefine a role map for the imported LDAP users.

However, if you have checked Enable Auto-Import of Users from LDAP Server, users will automatically be imported into JReport Server when they log in for the first time. The Enable LDAP Support and Enable Auto-Import of Users from LDAP Server options in the Configuration > LDAP > Server tab work together. The former determines whether an imported LDAP user can be used in JReport Server, and the latter determines whether LDAP users can be imported automatically, as shown in the following table:

☑=Checked; ☐=Unchecked

	Enable LDAP Support	**Enable Auto-Import of Users from LDAP Server**	**Can be used**
Local User	☑	☑	YES
	☑	☐	YES
	☐	☑	YES
	☐	☐	YES
Imported LDAP User	☑	☑	YES
	☑	☐	YES
	☐	☑	NO
	☐	☐	NO
None-Imported LDAP User	☑	☑	YES
	☑	☐	NO
	☐	☑	NO
	☐	☐	NO

Using LDAP server security information via the LDAP implementation of the Security API

JReport Server can access an LDAP server directly using the LDAP Security API implementation. To achieve this, you will need to turn on the LDAP security providers. There are three approaches to achieve this:

- **Configuring on the JReport Administration page**

  You must be a member of the administrators role in order to access the JReport Administration page. To use the LDAP security providers, on the JReport Administration page, click **Configuration** > **LDAP** > **Server**, then check **Enable Direct Authentication to LDAP Server**.

- **Configuring by API method**

  Invoke the following two methods in the interface ConfigurationLDAPServer:

```
/**
* Specifies whether or not the Server security system uses the LDAP providers.
* @return Returns true if using none imported LDAP support, otherwise returns false.
*/
public boolean isEnableNoneImportedLDAPSupport();
/**
* Specifies whether or not the Server security system uses none imported LDAP support.
* @param isEnableNoneImportedLDAPSupport Set this parameter to true if the Server
security system uses none imported LDAP support, otherwise set it to false.
*/
public void setEnableNoneImportedLDAPSupport (boolean isEnableNoneImportedLDAPSupport);
```

- **Editing the LDAP XML configuration file**

  Specify the following property in the LDAP XML configuration file:

  `<env-enableNoneImportedLDAPSupport>true</env-enableNoneImportedLDAPSupport>`.

  If the value is true, JReport Server security system will then use the LDAP providers. The default value of this property is false.

Note: In order to use LDAP providers, a valid admin user is required to manage the JReport Server. The following are rules for checking whether or not a user is an admin user:

- Whether or not the user is a member of the LDAP admin group. The LDAP admin group is a configuration option in the LDAP configuration XML file.
- Whether or not the user is a member of the administrators role. The user can be granted the role by a role map.

A user that meets one of these two rules is regarded as an admin user, and is thus allowed to access the JReport Administration page.

Troubleshooting LDAP configuration

If you encountered any problems during LDAP configuration, refer to the following for help.

LDAP configuration failure resulting in re-login failure as an admin user

An admin user may fail to carry out LDAP configuration properly, and thus then cannot log onto the JReport Administration page to manage the server. If this happens, you should follow the below steps:

1. Modify the property in the LDAP configuration XML file LDAPProperties.xml in `<install_root>\properties` as follows to turn off the Enable Direct Authentication to LDAP Server option:

   `<env-enableNoneImportedLDAPSupport>false</env-enableNoneImportedLDAPSupport>`

2. Restart JReport Server and log in as a built-in security admin user to correct the LDAP configuration.

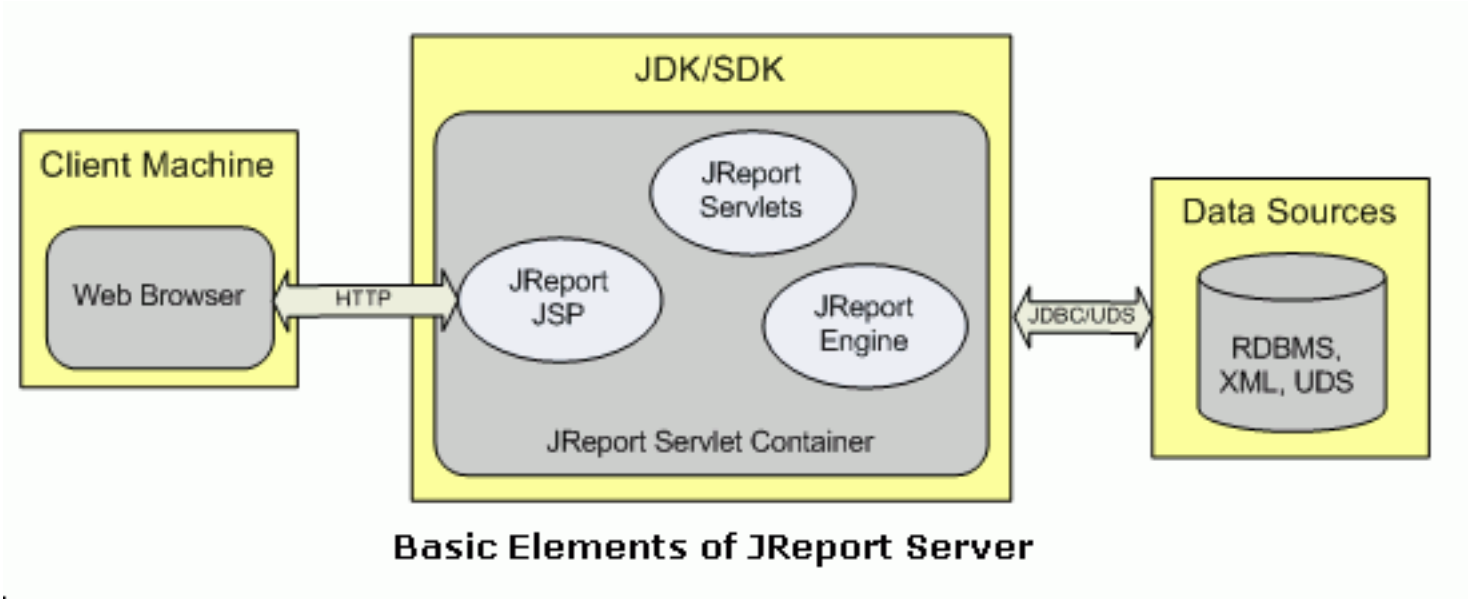
Warning messages in the advent of incorrect LDAP configuration

Apart from the notes offered on the JReport Administration page > Configuration > LDAP panel, prompt warning information is also provided in order to cope with incorrect LDAP configuration. The server system will prompt warning messages in the following cases:

- If you do not fill in the Admin Group field or specify an admin group.
- If the admin group specified does not hold a user.
- If the admin group specified does not exist in the LDAP server.

Security for accessing web pages

JReport Server provides a working web application as part of the product mix. This web application is implemented as a group of JSP pages and servlets that run in a Java web server's servlet container.



JReport Server controls access to these JSP pages and servlets by requiring that a user be logged in to a web session before the JSP pages or servlets will run.

JReport Server uses a security framework that allows for Single Sign On, so that a user only needs to login to an existing application and not need to do another login to access JReport Server web pages.

- Web page security
- Single Sign On

Web page security

JReport Server provides a working set of JSP pages and servlets that form a web application to schedule and run reports, and view results. This web application is only available to registered users who identify themselves and log in to JReport Server.

JReport Server provides an administrative function where user information is maintained. An administrator can register users with JReport Server, set passwords for them, and associate them with groups and roles. This database of user information is used to validate the user name and password that is used to log in to the web application.

Only logged-in users may access web pages

Only users who have logged in to the web application and created a session can access the web pages. The session is a state that is maintained across multiple HTTP Requests using the Jave EE servlet container's service to manage the session, and pass it into the JSP and servlet with each HTTP Request.

Each JSP page and servlet in the web application starts off with code that checks if the current HTTP Request is from a user who is already logged in. If it is, the rest of the code executes because this user is a known user. There may be more permissions checks within the code to control the specific request, but the user is allowed to access the web page.

If the current HTTP Request is not from a user who is already logged in, then the code attempts to log in the user using information available in the current HTTP Request.

If it is not possible to log in a user, then the JSP code recognizes that it is an Unauthorized request and does not allow the rest of the code to execute. It may send back an Unauthorized message (HTTP 401) to the browser or may do something else depending on configuration settings.

JRepor Server provides multiple ways for a user to log into the JReport web application.

- Use an external service to take care of the login.
- Use HTTP Authentication
- Use a proprietary protocol based on query parameters in the URL.

The JReport Server web pages does the login process in the order shown, but this discussion will cover them in reverse order because it is a better sequence to introduce the topics.

Using query parameters

When there is no current user logged in to the web session, JReport Server looks for specific query parameters that it has designated as a way to pass in login credentials in the URL. This is a proprietary login protocol developed to make it easy to identify a user by passing along the login request in the URL along with the report operation request. A set of query parameters are defined that are used to pass in the login credentials for a user.

If these query parameters exist, JReport Server will use their value and try to login the identified user to the session. If the credentials are valid, then the user is logged in to the session, and the web page can be accessed. Later HTTP Requests from that user will show that the user is logged in to the session.

Details about these parameters can be found in the javadoc entry for HttpUtil.checkLogin(), which is the method that does the checking.

- jrs.auth_id - value is the user name
- jrs.auth_pwd - value is the password
- jrs.authorization - value is string holding both user name and password encoded following a defined process

Here is an example URL requesting the JSP page viewVersion.jsp with credentials passed in to log in user "scott" with password "tiger".

```
http://localhost:8888/jinfonet/viewVersion.jsp?jrs.cmd=jrs.view_ver_def&jrs.
rst_version=1&jrs.ver_suff=.pdf&jrs.report=/InvoiceReport.cls?jrs.auth_id=scott?jrs.
auth_pwd=tiger
```

If no user is currently logged-in to the session, and the user "*scott*" with password "*tiger*" were a registered JReport user, then the viewVersion.jsp page would run and the user *scott* would become logged in to the session.

Using Http authentication

When there is no current user logged in to the web session, JReport Server looks at the HTTP Request header for the Authorization field. This is an element that follows standard HTTP protocol for HTTP Requests. Typically, it is filled in by the browser, based upon values entered by the user in a login dialog form presented by the browser when it receives a Unauthorized Request (HTTP 401) response to a URL request.

The browser responds to the Unauthorized Request response by presenting the user with the login dialog, then submitting the original URL request again (automatically), but this time with the HTTP Header having the Authorization field filled in with the user's credentials.

The browser remembers the user credentials and will send them in the HTTP Header Authorization for all subsequent requests to the same URL.

The JSP pages of JReport Server look in the HTTP Header Authorization field for user credentials and if found, will use them to validate the user. If the credentials validate a known user, that user is logged in to the web session, and the page requested by the URL runs. Later HTTP Requests from that user will show that the user is logged in.

As part of the HTTP Authentication protocol, JReport Server will respond to the browser when no user is logged in to the session. It sends back the Unauthorized Request (HTTP 401) response, expecting that the browser will gather credentials from the user and request the same HTTP Request again, with the HTTP Header now containing the Authorization field. The JSP page will see the Authorization field on the second time the request URL is sent.

HTTP Authentication is performed by this sequence of HTTP transmissions, and this processing by the server and browser.

1. Browser sends URL as a HTTP Request to access a JSP page.
2. JReport Server does not see a logged-in user, and passes back an Unauthorized Request message to the browser via an HTTP Response.
3. Browser prompts the user for login information.
4. Browser sends the URL again, this time with the HTTP Request header having the Authorization field holding the user credentials.
5. JReport Server does not see a logged-in user, but sees the Authorization field in the HTTP Request header, and attempts to login that user.
6. If JReport Server can validate the user credentials, the user is logged in to the session. If JReport Server cannot validate the user credentials, it sends back another Unauthorized Request message to the browser via an HTTP Response, and the cycle continues.
7. When the browser is asking the user for login information, the user may cancel the dialog. When this happens, the browser displays a message to the user showing that the request was unauthorized and does not send anything to JReport Server.

Note: The web page security system using HTTP Authentication and URL query parameters to pass in credentials requires a secure network connection (SSL) to be effectively secure. Over a non secure network, both the HTTP Authentication and URL query technique are lightweight security systems that should not be used in a production system.

Using external service

When there is no current user logged in to the web session, JReport Server will make a call to an external helper class method, asking it to return the User ID to use for the web session. If the method call returns a User ID, JReport Server will validate that the User ID is registered with JReport Server and if it is, it will log that user in.

This external helper class method is getExternalAuthorizedUser(). It is one of the methods defined in the Java interface HttpExternalAuthorized. This Java interface defines a class that developers can implement that will co-ordinate with the login protocol used by JSP pages in JReport Server as is described here.

When developers implement HttpExternalAuthorized and deploy it, it is registered with JReport Server. When this happens, it turns on what is called Single Sign On. This is a system that allows a local application to control how to log a user in, and to co-ordinate with JReport Server JSP pages so they do not enter into the HTTP Authentication dialog that would require a user to login to the system twice - once for the application and once for JReport Server.

The implementation of getExternalAuthorizedUser() can identify the user that is logged in to the application's web session and have JReport Server log that user in to the JReport Server web session automatically.

If no user is currently logged in to the application session, then the call to getExternalAuthorizedUser() will return a null. This tells JReport Server to work down the list of the other ways it has to login a user based on information in the HTTP Request. It will look in the HTTP Header, then it will look in the URL for query parameters. Eventually, if it fails to find any credentials it will recognize that the request for the web page is an Unauthorized Request.

At this point, it calls the HttpExternalAuthorized method >handleUnAuthenticatedRequst(). This method in the Java interface defines the function that can be implemented by developers to handle the case of a web page receiving an Unauthorized Request.

The implementation can generate an HTTP Response that redirects the browser to the application's login workflow and tell JReport Server to not engage in an HTTP Authentication dialog with the browser.

Validation of user during login

The explanation so far is about how login credentials are obtained to validate and login a user to the web session.

When the user name and password is obtained, JReport Server will validate if the login credentials match to a JReport Server user.

The built-in system for doing this looks at the information for the set of users that were registered with JReport Server using the administrative tasks for managing users. It validates that the password is correct for the given username based upon the information inside JReport Server's user database.

Application provides validation of user during login

JReport Server defines the AuthenticationProvider Java interface that developers can implement that will do validation that a password is correct for the given username, using the application's user database.

The developers can implement an instance of the AuthenticationProvider Java interface and register it with JReport Server. Then this class and its method isValidUser() will be used to validate the user instead of using the built-in validation method.

An example of an implementation of the AuthenticationProvider Java interface is available in the sample source area in the file DemoAuthenticationProvider.java in `<install_root>\help\samples\APISecurity`.

Limiting actions based on permissions

Once the JSP page is deemed accessible following the rules for login and validation, the JSP page is allowed to run. The JSP page is run when an JReport user is correctly logged in to the web session. However, it may be that the logged in user does not have permission to do the requested operation or have rights for the target resource of the operation. The user may not have permission to run a report, or the user may have permission to run a report, but not the specified one in the specified catalog.

This aspect of whether the user is authorized for the request is determined by JReport Server evaluating the information in the user database that was added by an administrative task to manage users and their permissions.

Application limits actions based on permissions

JReport Server defines the AuthorizationProvider Java interface that developers can implement that will replace the built-in system for evaluating permissions and determining authorization.

The developers can implement an instance of the AuthorizationProvider Java interface and register it with JReport Server. Then this class and its methods checkPrivilege() and isPermissionOK() will be used to determine privilege instead of using the built-in authorization methods.

An example of an implementation of the AuthorizationProvider Java interface is available in the sample source area in the file DemoAuthorizationProvider.java in `<install_root>\help\samples\APISecurity`.

Demonstration and examples

There are two folders in the sample code area of the release that can be used to learn more about web page security.

One contains a set of JSP pages that can be run to try out various styles of accessing a JSP page to see how the login dynamics works. This is the folder `<install_root>\help\samples\APISecurity\LoginLogout`. The entry point to the set of JSP pages is loginIndex.jsp. Read the comments in the JSP page to understand how to run the demonstration.

The other contains two implementations of HttpExternalAuthorized and a set of JSP pages that can be used to learn more about Single Sign On. This is the folder `<install_root>\help\samples\APISecurity\SingleSignOn`. The entry point to the set of JSP pages is customIndex.jsp. Read the comments in the java files and the JSP page to understand how to run the demonstration.

Single Sign On

JReport Server's web pages are built to work with an existing web application. In particular, it is possible to set up the web server so that a user of the website can login to an existing web application and have that login grant them access to JReport web pages. This is called the Single Sign On feature.

This is done by developers implementing the class defined by the JReport Server Java interface HttpExternalAuthorized and telling JReport Server to use that implementation.

The implementation can be aware of the application's technique for managing login state in the servlet session. This code can tell JReport Server which user is logged in. The implementation can redirect the user to the application's login workflow if the request is not from a logged in user.

This system gives the user one spot in the application to login. A successful login there will allow the user to run JReport Server web pages without doing another login dialog.

JReport Server is told to use the local implementation of ExternalAuthorized in two ways.

- **Use the command line to define a system property that registers the class.**

  The system property jrs.httpExternalAuthorized is used to hold the name of the class that implements HttpExternalAuthorized.

  If the name of the class is DemoExternalAuthorized.java, then change the script file that starts up JReport Server to include the parameter string: `-Djrs.httpExternalAuthorized=DemoExternalAuthorized`.

  The following shows an example:

  1. Create a table by DDL as below in the database:

     ```
     CREATE TABLE jr_auth(auth_key varchar(64), auth_uid varchar(256) NOT NULL, PRIMARY KEY(auth_key))
     ```

     Insert testing data for an example:

     ```
     insert into jr_auth (auth_key, auth_uid) values('987654321', 'admin');
     ```

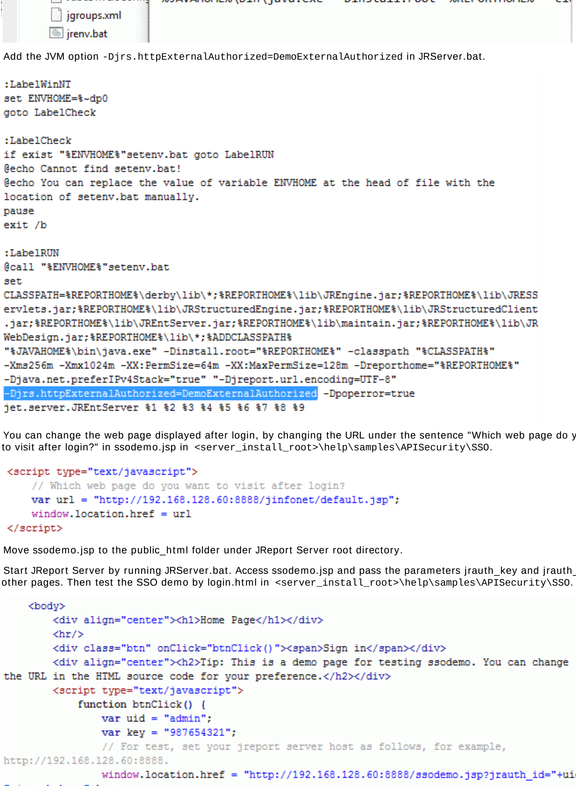
     Tip: jr_auth includes the inserted/updated data. auth_uid should be a JReport Server user name (the login user).

- Configure the database connection in databaseCfg.ini under the folder `<server_install_root>\help\samples\APISecurity\SSO`.

  ```
  driver=com.mysql.jdbc.Driver
  dbUrl=jdbc:mysql://IP_Address:3306/test
  dbUser=root
  dbPassword=1234
  ```

- Extract compile_tool.zip in `<server_install_root>\help\samples\APISecurity\SSO` to the same directory as DemoExternalAuthorized.java. Choose a compile tool according to your operating system, compile_tool.bat for Windows or compile_tool.sh for Linux. Configure the arguments in the corresponding file based on your environment. Then double-click the file to start compiling the Java code DemoExternalAuthorized.java.

- Modify the JReport Server start shell script JRServer.bat in `<server_install_root>\bin` to add JDBC driver and the compiled class of DemoExternalAuthorized.java as the first item in the class path.



- Add the JVM option `-Djrs.httpExternalAuthorized=DemoExternalAuthorized` in JRServer.bat.

  ```
  :LabelWinNT
  set ENVHOME=%~dp0
  goto LabelCheck

  :LabelCheck
  if exist "%ENVHOME%"setenv.bat goto LabelRUN
  @echo Cannot find setenv.bat!
  @echo You can replace the value of variable ENVHOME at the head of file with the
  location of setenv.bat manually.
  pause
  exit /b

  :LabelRUN
  @call "%ENVHOME%"setenv.bat
  set
  CLASSPATH=%REPORTHOME%\derby\lib\*;%REPORTHOME%\lib\JREngine.jar;%REPORTHOME%\lib\JRESS
  ervlets.jar;%REPORTHOME%\lib\JRStructuredEngine.jar;%REPORTHOME%\lib\JRStructuredClient
  .jar;%REPORTHOME%\lib\JREntServer.jar;%REPORTHOME%\lib\maintain.jar;%REPORTHOME%\lib\JR
  WebDesign.jar;%REPORTHOME%\lib\*;%ADDCLASSPATH%
  "%JAVAHOME%\bin\java.exe" -Dinstall.root="%REPORTHOME%" -classpath "%CLASSPATH%"
  -Xms256m -Xmx1024m -XX:PermSize=64m -XX:MaxPermSize=128m -Dreporthome="%REPORTHOME%"
  -Djava.net.preferIPv4Stack="true" "-Djreport.url.encoding=UTF-8"
  -Djrs.httpExternalAuthorized=DemoExternalAuthorized -Dpoperror=true
  jet.server.JREntServer %1 %2 %3 %4 %5 %6 %7 %8 %9
  ```

- You can change the web page displayed after login, by changing the URL under the sentence "Which web page do you want to visit after login?" in ssodemo.jsp in `<server_install_root>\help\samples\APISecurity\SSO`.

  ```
  <script type="text/javascript">
      // Which web page do you want to visit after login?
      var url = "http://192.168.128.60:8888/jinfonet/default.jsp";
      window.location.href = url
  </script>
  ```

- Move ssodemo.jsp to the public_html folder under JReport Server root directory.

- Start JReport Server by running JRServer.bat. Access ssodemo.jsp and pass the parameters jrauth_key and jrauth_id from other pages. Then test the SSO demo by login.html in `<server_install_root>\help\samples\APISecurity\SSO`.

  ```
      <body>
          <div align="center"><h1>Home Page</h1></div>
          <hr/>
          <div class="btn" onClick="btnClick()"><span>Sign in</span></div>
          <div align="center"><h2>Tip: This is a demo page for testing ssodemo. You can change
  the URL in the HTML source code for your preference.</h2></div>
          <script type="text/javascript">
              function btnClick() {
                  var uid = "admin";
                  var key = "987654321";
                  // For test, set your jreport server host as follows, for example,
  http://192.168.128.60:8888.
                  window.location.href = "http://192.168.128.60:8888/ssodemo.jsp?jrauth_id="+uid+
  "&jrauth_key="+key;
              }
          </script>
      </body>
  </html>
  ```

  Tip: Do not modify the two parameter names when using ssodemo.jsp we provided, or there will be exceptions. If you need to modify the parameter names, change them in all the three files: login.html, ssodemo.jsp, and DemoExternalAuthorized.java.

- **Use a Java API method call to register the class.**

  The Java API class HttpUserSessionManager has a method for setting the ExternalAuthrized object that JReport Server uses.

  If the name of the package is com.mycorp.myHttpExternalAuthorized, then in a JSP page, connect to JReport Server, then pass an instance of the class object for myHttpExtneralAuthorized as the parameter in the method HttpUserSessonManager.setHttpExternalAuthorized().

  ```
  <%@ page import="com.example.MyHttpExternalAuthorized" %>

          // initialize and connect to JReport Server
      initEnv(System.getProperties());
      HttpRptServer httpRptServer = HttpUtil.getHttpRptServer(request);

          // set the HttpExternalAuthorized object used by JReport Server
      httpRptServer.getHttpUserSessionManager().setHttpExternalAutorized(new myHttpExternalAuthorized());
  ```

There are examples of implementations of the ExternalAuthorized Java interface in the sample source files that come with JReport Server. Look in the folder `<install_root>\help\samples\APISecurity\SingleSignOn`. Read the comments in the source code for more information about Single Sign On and how the Java interface is used.

- samples\APISecurity\SingleSignOn\CustomHttpExternalAuthorized.java
- samples\APISecurity\SingleSignOn\com\example\MyExternalAuthorized.java

In that same SingleSignOn folder are several JSP pages that can be placed into the public_html\jinfonet folder and run as web applications to exercise and demonstrate how Single Sign On works. The file customIndex.jsp is the entry point page. It has comments inside it on how to run the demonstration.

Notes:

- The set of application users does not need to be the same set of users that are registered in JReport Server. However, the Single Sign On protocol does require that the application code identify the logged in user to JReport Server using a registered JReport Server user name.
- When the set of application users is different from the set of JReport Server users, then the application code for returning the logged in user will need a system to map the applications user to a JReport Server user.

Configuring JReport Server

Once JReport Server has been installed and you have prepared the reporting environment, you can then start it. However, you may find that JReport's default port number conflicts with an existing one in your computer, or that you need to generate log files for analyzing a problem. In these cases, you will need to configure your server. Before or while running the server, you can configure it according to your systems.

The configuration work can be performed in two ways: via the JReport Server UI or via configuration files. Some server UI options and properties in the configuration files are mapped and both function the same way. For these types of settings, you can take either way to do the configuration. Appendix 4: Mapping list of server UI options and properties details which server UI option are mapped to which properties in the configuration files.

This chapter focuses on the following topics:

- Configuration files
- Performing administrative configuration work on UI
- Configuring the server database
- Configuring dynamic connections
- Configuring another connection as a substitute for the catalog connection
- Configuring connection pool
- Configuring logs
- Changing the server context path

Configuration files

The following table lists the files provided for configuration purpose and the corresponding server UIs they are mapped to if applicable:

File	Description	Mapped Server UI	Related Document
<install_root>\bin\classes.properties	Apply old version Security API (UserAuthenticator)	-	-
<install_root>\bin \ConnectionPoolConfig.properties	Engine connection pool configuration	-	Configuring a connection pool
<install_root>\bin\datasource.xml	Connection mapping configuration	-	Configuring the datasource. xml file
<install_root>\bin\dbconfig.xml	Server DB configuration	JReport Administration page > Data > System DB/Realm DB/Profiling DB > Configuration	Configuring the server database in the dbconfig.xml file
<install_root>\bin\dhtml.properties	Configuration for the Action Task Manager	-	Action Task Manager > dhtml. properties
<install_root>\bin\faxconfig. properties	Configuration for sending fax	JReport Administration page > Configuration > Export > Fax	-
<install_root>\bin\jdbcdrivers. properties	Configure the JDBC drivers that can be auto loaded	-	-
<install_root>\bin \JdbcDriversConfig.properties	JDBC capability description	-	Configuring JdbcDriversConfig. properties
<install_root>\bin\LogConfig. properties	Configuration for log	JReport Administration page > Configuration > Log	Configuring logs
<install_root>\bin\mailconfig. properties	Configuration for sending e-mails	JReport Administration page > Configuration > E-mail Server JReport Administration page > Configuration > Export > E-mail	-
<install_root>\bin\server.list	Record relevant information about the clustered servers, including server name, server IP address, server RMI port, and server backup priority.	-	-
<install_root>\bin\server.properties	Comprehensive configuration	JReport Administration/ Console page	Appendix 1: Properties - server.properties file
<install_root>\properties \LDAPProperties.xml	Configuration for LDAP support	JReport Administration page > Configuration > LDAP > Server	-

Related topics:

- Appendix 4: Mapping list of server UI options and properties

Performing administrative configuration work on UI

JReport Server UI can be accessed remotely from a client machine through a web browser such as Internet Explorer. The JReport Administration page is available to administrators only. To perform administrative configuration work as an administrator, first log onto the JReport Administration page.

On the JReport Administration page, these tabs are entry to different-purposed configuration panels:

- Configuration - A comprehensive place for server configuration containing further-divided categories.
- Security - Where you manage user accounts and permissions.
- Profile - Where you define some initial settings for users' profiles.
- Cluster - Where you configure and manage a JReport Server cluster.
- Triggers - Where you create and manage triggers.
- Data - Where you manage the server data.
- Cube - Where you create and manage in-memory cubes for business views.
- Cache - Where you create and manage the caches of data, reports, security objects and images.

Configuring the server database

There are three databases in JReport Server: system, realm, and profiling. The system database holds resources of the global server scope, such as server.properties, global NLS, and so on. The realm database holds information of folders, nodes, versions, the security system, and the completed table. The profiling database holds server runtime related information. The realm database is necessary in order to run JReport Server. For best performance, you may want to configure the realm and profiling databases separately, depending on your environment.

When you install JReport Server, a dbconfig.xml file is automatically created in the directory `<install_root>\bin`. The database configuration information is stored in this configuration file. You can configure your database by using the dbconfig.xml file.

These databases have been tested workable as the server database: Apache Derby, HSQLDB, MySQL, Microsoft SQL Server, IBM DB2, Oracle, Sybase, and Informix.

This section presents the ways of configuring a server database for JReport Server in two different environments. Also, you can specify a table space in the database configuration for JReport Server to create tables in it.

- Configuring the server database in a standalone environment
- Configuring the server database when integrating with an application server
- Creating tables in a specified table space
- Schema support
- Configuring JdbcDriversConfig.properties

Notes:

- If you are using MySQL, make sure it is of version 5 or above.
- If your server database uses DB2 and the charset is DBK, there will be the exception of encoding not supported.
- When setting SQL 2000 as the server database, the driver should use jtds.jar, otherwise the scheduling feature cannot work.
- ODBC is not supported.

Related topics:

- Managing server data

Configuring the server database in a standalone environment

When JReport Server is running in a standalone environment, you can configure the database for it remotely on the JReport Administration page. Also, since the server database configuration information is stored in the dbconfig.xml file in the directory `<install_root>\bin`, you can also configure the server database in this file.

The following presents the two ways of configuring the server database in a standalone environment:

Configuring on the JReport Administration page

1. Log onto the JReport Administration page, click **Data** on the system toolbar and then select **System DB**, **Realm DB**, or **Profiling DB** from the drop-down menu according to your requirement.

   Note: The Profiling DB option is not shown by default on the drop-down menu. In order to make it shown, you should set the server.profiling.enable property to true in the server.properties file in the `<install_root>\bin` directory.

2. Select a realm if it is the Realm DB or Profiling DB panel.

3. In the Configuration tab, select a DBDriver from the Driver drop-down list and provide the driver class path information in the Driver Class Location field.

   Notes:

   - For HSQLDB and Derby databases, you do not need to specify the Driver Class Location. For all other databases, you will have to provide the driver class path information unless it has already been added to the class path of setenv.bat (setenv.sh on Unix) file during installation or by editing the setenv.bat file.

   - For Microsoft SQL Server 2005 and later versions, use the driver com.microsoft.sqlserver.jdbc.SQLServerDriver, otherwise use com.microsoft.jdbc.sqlserver.SQLServerDriver for versions earlier than Microsoft SQL Server 2005.

4. Type a valid URL that can be used to establish a connection to the database. The valid format of the URL should be provided by the DBDriver vendor.

5. Provide the user ID and password.

6. To test the connection, click **Test**. To update the database configuration and to apply the settings, click **Update** and then restart the server to finalize the function.

Configuring in the dbconfig.xml file

In dbconfig.xml, you can configure the server database using one of two methods. One is to specify the URL, driver, user and password individually. This method of configuration can be modified through the JReport Administration page. For example:

```
<database name="systemtables/realmtables/profile">
    <url>...</url>
    <driver>...</driver>
    <user>...</user>
    <password>...</password>
</database>
```

Note: The `<user>` and `<password>` information is encrypted. The `<user>` and `<password>` tags will be replaced by the `<encrypt-sign>` tag after JReport Server's startup as follows:

```
<encrypt-sign>enDkq7srM9cHhoUwzYXJ3NvcDIYk</encrypt-sign>
```

If you want to change user or password, delete the `<encrypt-sign>` tag and add the `<user>` and `<password>` tags in the dbconfig.xml file.

The other is to use the `<datasource>` tag. For example:

```
<database name="systemtables/realmtables/profile">
    <datasource>
        jdbc://user:password@jdbc:odbc:jreport-realmtables#driver=sun.jdbc.odbc.JdbcOdbcDriver
    </datasource>
</database>
```

Here are two examples for your reference:

- The following example is for Oracle. Modify the dbconfig.xml file as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<dbconfig>
    <workspace name="systemRealm">
        <database name="systemtables">
            <url>jdbc:oracle:thin:@dbhost:1521:SystemDB</url>
            <user>test</user>
            <password>1234</password>
            <driver>oracle.jdbc.OracleDriver</driver>
        </database>
    </workspace>
    <workspace name="defaultRealm">
        <database name="realmtables">
            <url>jdbc:oracle:thin:@dbhost:1521:RealmDB</url>
            <user>test</user>
            <password>1234</password>
            <driver>oracle.jdbc.OracleDriver</driver>
        </database>
    </workspace>
</dbconfig>
```

- The following example is for a DataDirect driver. Modify the dbconfig.xml file as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<dbconfig>
    <workspace name="systemRealm">
        <database name="systemtables">
            <url>
                jdbc:datadirect:sqlserver://dbhost:1433;DatabaseName=SystemDB
            </url>
            <user>test</user>
            <password>1234</password>
            <driver>
                com.ddtek.jdbc.sqlserver.SQLServerDriver
            </driver>
            <dbtype>Microsoft SQLServer</dbtype>
        </database>
    </workspace>
    <workspace name="defaultRealm">
        <database name="realmtables">
            <url>
                jdbc:datadirect:sqlserver://dbhost:1433;DatabaseName=RealmDB
            </url>
            <user>test</user>
            <password>1234</password>
            <driver>
                com.ddtek.jdbc.sqlserver.SQLServerDriver
            </driver>
            <dbtype>Microsoft SQLServer</dbtype>
        </database>
    </workspace>
</dbconfig>
```

Notes:

- Usually, JReport Server automatically creates database tables the first time it is started. The database information that JReport Server uses is defined in the dbconfig.xml file. However, if the user ID defined in this file does not have the permission to create tables in the database, JReport Server will fail to complete the operation. In this case, you will need another user, such as the database administrator (who holds the relevant permissions), to create a set of empty tables in the user's schema using the provided SQL files. These SQL files can be found in `<install_root>\script_files`.

- For MySql 5.0.xx, the letters in user name should all be lowercase.

- Because of the compatibility between the third package Quartz used by JReport Server and the driver of Oracle database, the Quartz package will throw exception when the class path of oracle database driver is added into dbconfig.xml. To avoid this problem, when the JReport server works with Oracle database, do not set database driver class path in dbconfig.xml.

Configuring the server database when integrating with an application server

JReport Server supports connecting an RDBMS to access its system data via JDBC. The JDBC configuration information is stored in the file dbconfig.xml in `<install_root>\bin`. You can create a database connection according to this configuration file. Also, with the Java EE Data Source Support feature, in a Java EE environment, JReport Server can get the predefined javax.sql.DataSource by JNDI APIs.

Here, the term dsInfo is used to indicate where JReport Server can obtain the JDBC connection information. It is a key-value pair. The name and value for dsInfo are defined as follows:

- The name should be: jreport.datasource.`<dbname>`, where, `<dbname>` is JReport Server's inner database name. It must be systemtables, realmtables, or profile. For example, the dsInfo name is jreport.datasource.realmtables.
- The value should be in the form of a URL. There are three protocols supported by JReport Server:
  - `file:///absolute_path_of_config_file`

    For example: `file:///JReport/Server/bin/dbconfig.xml`
  - `jdbc://[<jdbc-user:jdbc-password>@]<jdbc-url>[#<attribute-name=attribute-value>,]`

    For example: `jdbc://user:password@jdbc:odbc:jreport-realmtables#driver=sun.jdbc.odbc.JdbcOdbcDriver`
  - `jndi://[<jdbc-user:jdbc-passoword>@]datasource_name[#<attribute-name=attribute-value>,]`

    For example: `jndi://jdbc/jreport-realmtables`

The dsInfo can be specified in several places or levels, such as VM properties, ejb-jar.xml, web.xml and `<install_root>\bin\dbconfig.xml`. The sequence to load the information is:

VM system environment > ejb-jar.xml (EAR mode) > web.xml (WAR mode) > dbconfig.xml.

Notes:

- The jreport.ear file that JReport Server creates does not specify any Java EE information. By default it will load from the dbconfig.xml file.
- There is a limitation in WebSphere. JReport Server's dsInfo can be configured to the WEB-INF/web.xml with JNDI name but not RESOURCE name. JBoss works fine for either one.
- Do not use reporthome, @ or # in the JNDI name or Resource name.

The following shows specifying the dsInfo in web.xml, ejb-jar.xml, and dbconfig.xml respectively:

Specifying data source in web.xml (for the WAR mode)

When integrating the JReport Server in a WAR package, you can specify the dsInfo in the WEB-INF/web.xml file using the `<env-entry></env-entry>` or `<context-param></context-param>` tags. However, the use of the `<env-entry></env-entry>` tags is the recommended way since the tags are also supported in ejb-jar.xml (if you call the Server API in your EJB).

The following is an example of specifying the dsInfo in WEB-INF/web.xml using the `<env-entry></env-entry>` tags:

```
<web-app>
    ...

    <env-entry>
        <env-entry-name>jreport.datasource.realmtables</env-entry-name>
        <env-entry-value>jndi://java:/resource-name</env-entry-value>
        <env-entry-type>java.lang.String</env-entry-type>
    </env-entry>
</web-app>
```

The following is an example of specifying the dsInfo in WEB-INF/web.xml using the `<context-param></context-param>` tags:

```
<web-app>
    <context-param>
        <param-name>jreport.datasource.realmtables</param-name>
        <param-value>jndi://datasource_name_which_is_predefined</param-value>
    </context-param>
    ...

</web-app>
```

Specifying data source in ejb-jar.xml or web.xml (for the EAR mode)

When the JReport Server is used with APIs by EJBs, you can specify the dsInfo either in the file ejb-jar.xml in `/META-INF` or web.xml in `/WEB-INF`.

For information about specifying a data source in the web.xml file, see section [Specifying data source in web.xml (for the WAR mode)](#).

Note: If the dsInfo is stored in a Web module, you should add the JReport Server context listener to the WEB-INF/web.xml file. Here is an example:

```
<web-app>
    <listener>
        <listener-class>
             jet.server.servlets.JRServerContextListener
             </listnener-class>
    </listener>
    ...
</web-app>
```

You can specify the dsInfo in the META-INF/ejb-jar.xml file using the `<env-entry></env-entry>` tags. For example, here is an EJB named firstEJB which creates the JReport Server instance. You can add the `<env-entry></env-entry>` tags to the EJB's configuration file - META-INF/ejb-jar.xml as follows:

```
<ejb-jar>
    <enterprise-beans>
        <session>
            <ejb-name>firstEJB</ejb-name>
            ...
            <env-entry>
                <env-entry-name>jreport.datasource.realmtables</env-entry-name>
                <env-entry-value>jndi://java:/resource_name</env-entry-value>
                <env-entry-type>java.lang.String</env-entry-type>
            </env-entry>
        </session>
        ...

    </enterprise-beans>
    ...

</ejb-jar>
```

Specifying data source in dbconfig.xml

By default, JReport Server assumes that jreport.datasource.realmtables=file:///${rpthome}/bin/dbconfig.xml. You can specify either a JDBC or JNDI data source using the `<datasource></datasource>` tags in the dbconfig.xml file.

For details about configuring the JDBC data source, see [Configuring the server database in a standalone environment](#).

The following is an example of specifying the JNDI data source:

```
<?xml version="1.0" encoding="UTF-8"?>
<dbconfig>
    <workspace name="systemRealm">
        <database name="systemtables">
            <datasource>jndi://datasource-name</datasource>
        </database>
    </workspace>
    <workspace name="defaultRealm">
        <database name="realmtables">
            <datasource>jndi://datasource-name</datasource>
        </database>
    </workspace>
</dbconfig>
```

Creating tables in a specified table space

JReport Server supports creating tables in a user-specified table space in a database that supports table spaces, such as DB2 and Oracle. A key-value pair tablespace is provided to specify a table space into which JReport Server will create database tables. This key-value pair is then passed to JReport Server through the JDBC configuration. JReport Server retrieves the table space information from the JDBC (data source) configuration, and then creates tables in the specified table space.

Tablespace can be configured either in the dbconfig.xml file by using the `<tablespace></tablespace>` tags, or in dsInfo by adding the attribute tablespace=table_space_name.

In dbconfig.xml

Add the `<tablespace></tablespace>` tags in the dbconfig.xml file as follows:

```
...
<workspace name="defaultRealm">
    <database name="realmtables">
        <driver classpath="...">jdbc_driver_name</driver>
        <url>jdbc_url</url>
        <user>jdbc_user</user>
        <password>jdbc_password</password>
        <tablespace>table_space_name</tablespace>
    </database>
</workspace>
...
```

In dsInfo

Since the dsInfo supports JNDI and JDBC protocols, you can add the attribute tablespace=table_space_name in the JNDI or JDBC statement.

- jdbc://[<jdbc-user>:<jdbc-password>@]<jdbc-url>[#<attribute-name=attribute-value>,]

  For example:

  `jdbc://user:password@jdbc:odbc:jreport-realmtables#driver=sun.jdbc.odbc.JdbcOdbcDriver,tablespace=myTablespace`

- jndi://[<jdbc-user>:<jdbc-password>@]<datasource-name>[#<attri-name=attri-value>,]

  For example:

  `jndi://jdbc/jreport-realmtables#tablespace=myTablespace`

Schema support

JReport Server supports DBMS schemas in order to work well with DBMSs that are currently supported by JReport Server and that support schemas. Those DBMSs include Oracle, DB2, SQL Server, and Sybase. You can specify schema information either in the dbconfig.xml file using the <schema></schema> tags, or in dsInfo by adding the attribute schema=schema_name.

Note: For Sybase, the schema name must use capital letters, for example, ABCDE.

In dbconfig.xml

Add the <schema></schema> tags in the dbconfig.xml file as follows:

```
...
<workspace name="defaultRealm">
    <database name="realmtables">
        <driver classpath="...">jdbc_driver_name</driver>
        <url>jdbc_url</url>
        <user>jdbc_user</user>
        <password>jdbc_password</password>
        <schema>schema_name</schema>
    </database>
</workspace>
...
```

In dsInfo

Since the dsInfo supports JNDI and JDBC protocols, you can add the attribute **schema=schema_name** in the JNDI or JDBC statement. The JNDI approach is only available when deploying JReport Server into a Java Application Server, not when it is running in standalone mode.

- jdbc://[<jdbc-user>:<jdbc-password>@]<jdbc-url>[#<attribute-name=attribute-value>,]

  For example:

  jdbc://user:password@jdbc:odbc:jreport-realmtables#driver=sun.jdbc.odbc.JdbcOdbcDriver,schema=db2admin

- jndi://[<jdbc-user>:<jdbc-password>@]<datasource-name>[#<attri-name=attri-value>,]

  For example:

  jndi://jdbc/jreport-realmtables#schema=db2admin

Configuring JdbcDriversConfig.properties

This document describes how to configure the file JdbcDriversConfig.properties in the `<install_root>\bin` directory for better performance.

Limiting the size of the fetch data buffer

If you are using a database which supports the JDBC method Statement.setFetchSize(), you can request the database retrieve a specified number of rows in each read instead of all rows which minimizes memory usage which can improve performance. It can also avoid Java heap out of memory errors when doing large queries. JReport provides a property setFetchSize in the JdbcDriversConfig.properties file for you to do this.

The following is an example of setting the setFetchSize property in the JdbcDriversConfig.properties file and detailed description of each property:

```
DriverName_D1 = MySQL-AB JDBC Driver
Vendor_D1 = MySql.com
Version_D1 = 3.0.8-stable ( $Date: 2003/05/19 00:57:19 $, $Revision: 1.27.2.18 $ )
setFetchSize_D1 = 5000
```

DriverName_[Name]

Database driver name that should be used to establish a connection. You can use connection.getMetaData().getDriverName() to obtain the value.

Vendor_[Name]

The vendor of current JDBC driver. It is a string value.

Version_[Name]

The version of current JDBC driver. You can use connection.getMetaData().getDriverVersion() to obtain the value.

setFetchSize_[Name]

The number of rows retrieved in each buffer from the database. It is an Integer value that should be larger than 0 and less than or equal to getMaxRows(). The default value is different according to the driver in use. You can refer to your own database driver specification.

_[Name]

It is a user-defined name used to mark a group of driver settings different from the other groups.

Specifying where to implement the maximum query run time and number of records

In the JdbcDriversConfig.properties file, you can also specify where the properties, Maximum Rows and Maximum Duration will be implemented, on JReport side or on the database side.

The following is an example of setting the two properties:

```
DriverName_D2 = Oracle JDBC driver
Vendor_D2 = Oracle
Version_D2 = 9.2.0.3.0
supportMaxRowPushDown_D2 = true
supportMaxDurPushDown_D2 = true
```

DriverName_[Name]

Database driver name that should be used to establish a connection. You can use connection.getMetaData().getDriverName() to obtain the value.

Vendor_[Name]

The vendor of current JDBC driver. It is a string value.

Version_[Name]

The version of current JDBC driver. You can use connection.getMetaData().getDriverVersion() to obtain the value.

SupportMaxRowPushDown_[Name]

Specifies where the property Maximum Rows will be implemented, on JReport side or on the database side. By default, it is set to true, which means the property will take effect on the database side. If it is set to false, the property will take effect on JReport side.

SupportMaxDurPushDown_[Name]

Specifies where the property Maximum Duration will be implemented, on JReport side or on the database side. By default, it is set to true, which means the property will take effect on the database side. If it is set to false, the property will take effect on JReport side.

_[Name]

It is a user-defined name used to mark a group of driver settings different from the other groups.

Canceling running query

If your JDBC driver support the Cancel Running Query feature and you want JReport to cancel the running query used by a report in the database when you cancel the running task of the report, for example, when you click the Cancel button on the report processing page of Page Report Studio, or when you choose to stop a report running in background mode, configure the file JdbcDriversConfig.properties as follows:

1. Add the JDBC driver you use into JdbcDriversConfig.properties if it is not listed there.
2. In the file JdbcDriversConfig.properties there is a line:

   `#supportCancelQueryStatement_D5=true`

   Remove the sign # and use your driver number instead of D5 to activate the feature.
3. Restart your JReport Server, then the next time, when you choose to cancel a running task, the query used by the report will also be cancelled in the database, provided that the DBMS supports this action.

Notes:

- After you modified the JdbcDriversConfig.properties file, you need start the server to load the changes.
- If there are more than one group with the same group marking name, the last group will be adopted.
- If the sign # is seen before "DriverName" of a group, or if setFetchSize is given a negative value, the whole group will be disabled.

Configuring dynamic connections

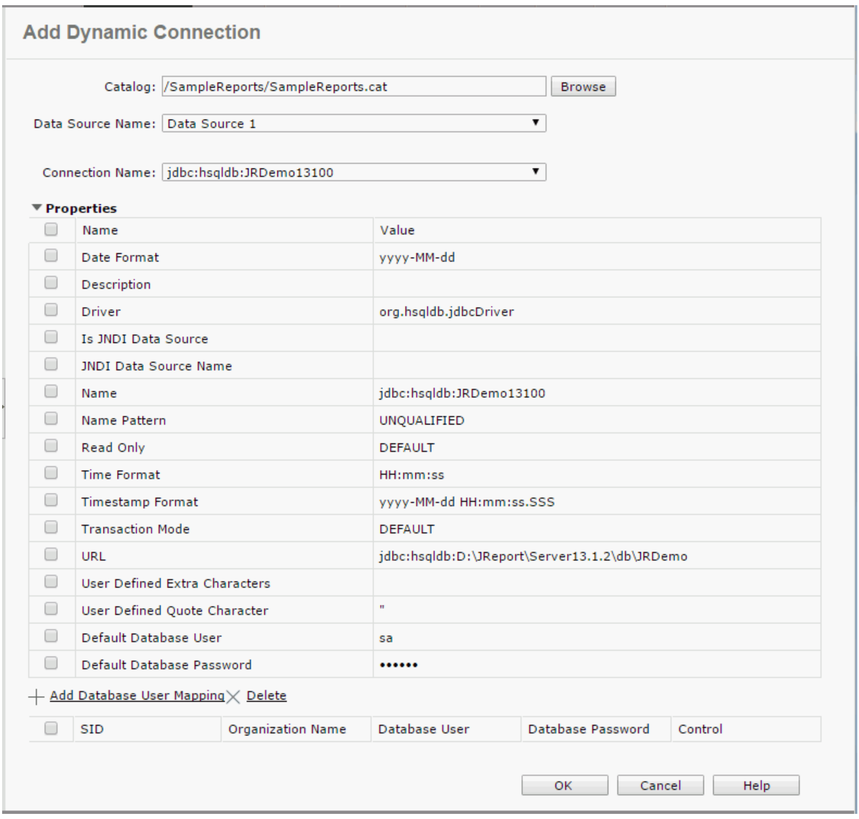
A dynamic connection is a group of connection properties that can be used to override the original data source connections in a catalog. The server administrator can modify some of the original catalog connection properties, and then save these modified properties as a dynamic connection property. For a dynamic connection, the administrator can define a database user mapping table which is a table containing the columns: SID, database user, and database password. JReport Server manages the dynamic connections in the server system database.

When running a report or scheduling a report to run, if there are multiple dynamic connections available to the log-in user, the user needs to select one. Then the server will pass the changed connection properties of the selected dynamic connection into the report engine. The report engine will merge the changed properties with the other original catalog connection properties to setup the database connection.

Dynamic connections are categorized at the organization level. If the system administrator creates a dynamic connection, the connection maps to no organization. If an organization administrator creates a dynamic connection, the connection maps to the organization. If a dynamic connection is defined in an organization, all users/groups/roles must be from the same organization. If a connection is defined in non-organization, all users/groups/roles must be from non-organizations. System administrator can define, view and edit all the dynamic connections.

Creating a dynamic connection

1. On the JReport Administration page, click **Configuration** on the system toolbar and then select **Dynamic Connections** from the drop-down menu.
2. In the Dynamic Connections panel, click the **Add Dynamic Connection** link. The Add Dynamic Connection dialog appears.



3. Click **Browse** to select a catalog in the Select Catalog dialog.
4. From the Data Source Name drop-down list, select a data source available in the selected catalog.
5. From the Connection Name drop-down list, select a connection to connect to the data source.
6. Click **Properties** to expand the properties table. To change a property's value, first select the checkbox beside it, and then enter a value in the text box or select a value from the drop-down list as required.
7. Click **Add Database User Mapping** to define the mapping relationship between SID and database user/password. A mapping row will be added in the mapping table. From the SID drop-down list, select a user, role, or group name. Double-click in the text box of Database User and Database Password and you can then type in database user name and password. Then click **Test Connection** to test whether the user name and password can connect to the database.

   Within a dynamic connection, make sure that an SID is mapped to only one pair of database user and name.
8. Click **OK** to create the dynamic connection which will be listed in the Dynamic Connections panel.

Managing dynamic connections

In the Dynamic Connections panel, the administrator can perform the following actions:

Searching for a dynamic connection

First select a search category from the drop-down list, then type the text in the text box and click to start the search. The **x** button in the text box is used to clear the input text. After the results are filtered, to return to the state before search, click **x** in the text box.

There are the following search categories:

- **Catalog**
  Searches for catalogs.
- **Connection Name**
  Searches for connection names.
- **SID**
  Searches for SIDs.
- **Database User**
  Searches for database users.

For example, in order to find the dynamic connections which use the catalog SampleReprts.cat, you can select Catalog from the search category drop-down list and input *sam* as the keyword. JReport will search for *sam* in the Catalog column and the dynamic connections whose catalogs contain the text will be kept in the dynamic connection table.

Editing a dynamic connection

Click the connection name to open the Edit Dynamic Connection dialog, in which you can edit the properties of the dynamic connection as you want.

Deleting dynamic connections

Select the checkboxes ahead of the dynamic connections and then click **Delete**. To delete all the dynamic connections at a time, select the checkbox on the column header and then click **Delete**.

Applying dynamic connections

One user can have multiple dynamic connections for one catalog data source. When running a report, if there are multiple dynamic connections for the log-in user:

- For advanced run, the user needs to select a dynamic connection from the available ones.
- For schedule run, if the selected dynamic connection cannot be found when running reports, then the first available dynamic connection will be applied.
- For direct run, link reports, API run, getting parameter values from the database, and getting filter values from the database, the first available dynamic connection is applied automatically.

Connection priority

The priority from high to low is: dynamic connection > the connection specified in datasource.xml > catalog connection.

Related topics:

- Dynamic connection API

Configuring another connection as a substitute for the catalog connection

Normally, after a report has been created, it is fixed to a specific catalog connection. With the Java EE Data Source Support feature, you can change to another runtime JDBC or JNDI data source to run reports. This allows you to control the catalog data source connections in JReport Server to connect dynamically to your production data sources.

This section focuses on the following:

- Configuring the datasource.xml file
- The connection priority
- Reloading connection information from datasource.xml
- Integrating JReport Server with your Java application server
- Example 1: Using JNDI data source connections of WebLogic Server
- Example 2: Using JNDI data source connections of JBoss server

Configuring the datasource.xml file

In order to enable you to conveniently set different connections in JReport Server, a configuration file, datasource.xml, is provided with which you can define the connection to use at runtime by setting up connection mapping information such as the JDBC driver, URL, or JNDI data source name, user and password, depending on the data source your application uses.

The datasource.xml file is located in the `<install_root>\bin` directory. Change this file according to the data source you want to use. Two types of connections are supported by JReport Server. They are JNDI data source connection and JDBC connection.

There can be multiple `<datasource></datasource>` tags. Each pair will map one of the connections in your catalog. You can define mapping connections for multiple connections for more than one catalog. The catalog connection name should be unique in the datasource.xml file, otherwise the latter `<datasource>` information will overwrite the previous ones.

The content of the datasource.xml file is as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<datasource-mapping>
    <datasource>
        <catalog-connection-name>Connection1</catalog-connection-name>
        <connection-type>JNDI</connection-type>
        <jndi-datasource>Sample</jndi-datasource>
    </datasource>
    <datasource>
        <catalog-connection-name>Connection2</catalog-connection-name>
        <connection-type>JDBC</connection-type>
        <driver>sun.jdbc.odbc.JdbcOdbcDriver</driver>
        <url>jdbc:odbc:jinfonet</url>
        <user>Username</user>
        <password>Password</password>
    </datasource>
</datasource-mapping>
```

Element descriptions

- **catalog-connection-name**
  Specifies the name of the catalog connection that you want to substitute. Note that this is the connection name that you can see in JReport Catalog Browser. A unique connection name in a catalog is suggested.

- **connection-type**
  Specifies the connection type, which can be either JNDI or JDBC.

- **jndi-datasource**
  If you are using a JNDI data source, specify the JNDI data source name you defined in your Java application server which you want to use as the substitute connection.

- **driver & url**
  If you are using a JDBC data source, specify the JDBC Driver and URL in these two attributes.

- **user & password**
  Specifies the user name and the password.

  The `<user>` and `<password>` information is encrypted. The `<user>` and `<password>` tags will be replaced by the `<encrypt-sign>` tag after JReport Server's startup as follows:

  ```
  <encrypt-sign>enDkq7srM9cHhoUwzYXJ3NvcDIYk</encrypt-sign>
  ```

  If you want to change user or password, delete the `<encrypt-sign>` tag and add the `<user>` and `<password>` tags in the datasource.xml file.

  You can choose whether to provide database user and password information for the JNDI connection, using the `<user>` and `<password>` tags. If the user and password information is provided in this file, it will be used to set up the connection regardless of the settings defined in your application server. Otherwise, the settings defined in your application server will be used. Specially, for WebLogic users, you should provide user name and password for WebLogic console instead of the database.

- **refresh-support-info**
  Optional. If it is true, JReport will get support information including reverse words, quote characters and so on from the driver each time fetching data from the database. If it is not set, the property will be treated as false. It is recommended that you set this property to true when connecting to a different database.

- **quote-character**
  Optional. Specifies the quote characters.

- **extra-characters**
  Optional. Specifies the extra characters.

- **date-format**
  Optional. Specifies the date format.

- **datetime-format**
  Optional. Specifies the datetime format.

- **time-format**
  Optional. Specifies the time format.

- **transaction-readonly**
  Optional. Specifies the transaction "read only" property. Possible value: default, Read only, or Read&Write. Ignore other values.

- **transaction-mode**
  Optional. Specifies the transaction mode. Possible value: default, None, Read Uncommitted, Read Committed, Repeatable Read, or Serializable. Ignore other values.

- **char-to-be-replaced**
  Optional. Specifies the characters that are to be replaced.

- **char-replaced-by**
  Optional. Specifies the characters used to replace the characters specified by char-to-be-replaced.

The connection priority

If you are using a catalog connection, datasource.xml, and at the same time, you have set new connection information at runtime via the Server API/URL ((jrs.jdbc_driver, jrs.jdbc_url, jrs.db_user, jrs.db_pswd), the priority of these three is as follows:

Runtime configuration of connection > datasource.xml > catalog connection

This means that if the approach with higher priority fails to get the connection, the one with the next lower priority will be used.

The runtime configuration of connection includes the connection object and connection info (such as url, driver, and user/pwd), and the connection object has higher priority than the connection info.

There are two ways of changing the connection by API at runtime, one is set the connection info by specifying the data source name, which will only change the connection of the indicated data source; the other is set the connection info without a specific data source name, which is regarded as a default change and will change the connections that have not been changed by the previous way.

Assuming that the catalog has two databases named as data source 1 and data source 2, here are some examples:

- A user does not set datasource.xml, but sets the connection by API without specifying a data source name. Both data source 1 and data source 2 will use the new setting.
- A user does not set datasource.xml, but sets a new connection A by specifying the data source name "data source 2". The result is: data source 1 has no change and data source 2 will use the connection A.
- A user does not set datasource.xml, but sets two connections by API, a new connection A by specifying the data source name "data source 2" and a new connection B without specifying a data source name. The result is: data source 1 will use the connection B and data source 2 the connection A.
- A user sets "data source 2" connection in datasource.xml and a new connection A by API by specifying the data source name "data source 2". The result is: data source 1 has no change and data source 2 will use connection A ignoring the setting in datasource.xml.
- A user sets "data source 2" connection B in datasource.xml and a new connection A by API without specifying the data source name. The result is: data source 1 will use the connection A and data source 2 the connection B.

Related topics:

- Switching report database connection in URL
- Opening web reports in Web Report Studio via JSON for switching connection for web reports via JSON
- Running dashboards via URL for switching connection for dashboards

Reloading connection information from datasource.xml

After you have made changes to the datasource.xml file, you need to reload it in JReport Server. This file can be reloaded using the following ways:

- Call jet.server.api.admin.ConnectionInfoProviderService.reloadFile().
- Log onto the JReport Administration page, go to **Configuration** > **Connection** panel, and then click the **Reload** button.
- Restart JReport Server.

Integrating JReport Server with your Java application server

After updating the datasource.xml file, you can integrate JReport Server with your application server.

If you are using a WAR file, make sure that the datasource.xml file is included in the WAR file. Before making the JReport Server WAR file, in makewar.xml in the `<install_root>\bin` directory, add `<include name="datasource.xml" />` into the `<zipfileset dir="${installroot}/bin" prefix="workspace/bin">` section. Before you deploy the WAR file to a web server, make sure that the datasource.xml file is included in the `jreport.war/WEB-INF/lib/jrenv.jar/workspace/bin` directory.

If you are integrating JReport Server to an application using a non-JReport WAR file, usually you need to specify the -Dreporthome parameter in the application server so that the application server can locate and load the resources it needs. Since the datasource.xml already exists in the bin directory, you do not need to do anything special for this file.

Example 1: Using JNDI data source connections of WebLogic Server

Using WebLogic as an example, take the following steps:

1. Start WebLogic server, navigate to <YourProjectName> > Services > JDBC > Connection Pools. In the Configuration tab, make sure that the connection you are going to use is there and correctly set up.
2. Browse to <YourProjectName> > Services > JDBC > Data Sources. In the Configuration tab, create a data source using the JNDI name MyJNDISample. Select the correct connection from the Pool Name drop-down list.
3. Open datasource.xml in the <install_root>\bin directory and add:

```
<datasource-mapping>
    <datasource>
        <catalog-connection-name>ConnectionName</catalog-connection-name>
        <connection-type>JNDI</connection-type>
        <jndi-datasource>MyJNDISample</jndi-datasource>
        <user>youruserid</user>
        <password>yourpassword</password>
    </datasource>
</datasource-mapping>
```

4. Build the JReport Server WAR file using the makewar utility in the <install_root>\bin directory.
5. Deploy JReport Server to WebLogic.
6. Run the report that uses this connection.

Example 2: Using JNDI data source connections of JBoss server

Using JBoss as an example, in this Example, DB2 is used, and JBoss is installed to `D:\jboss-4.0.1`, take the following steps:

1. Copy the DB2 driver to `D:\jboss-4.0.1\server\default\lib`.
2. Create a db2-ds.xml file in `D:\jboss-4.0.1\server\default\deploy`, and add the following content:

```
<?xml version="1.0" encoding="UTF-8"?>
<datasources>
    <local-tx-datasource>
        <jndi-name>DB2DataSource</jndi-name>
        <connection-url>jdbc:db2://dbs-b/sample</connection-url>
        <driver-class>com.ibm.db2.jdbc.net.DB2Driver</driver-class>
        <user-name>db2admin</user-name>
        <password>db2admin</password>
        <min-pool-size>0</min-pool-size>
        <metadata>
            <type-mapping>DB2</type-mapping>
        </metadata>
    </local-tx-datasource>
</datasources>
```

3. Modify datasource.xml in `<install_root>\bin`. Make sure that the following are included:

```
<datasource-mapping>
    <datasource>
        <catalog-connection-name>ConnectionName</catalog-connection-name>
        <connection-type>JNDI</connection-type>
        <jndi-datasource>java:/DB2DataSource</jndi-datasource>
    </datasource>
</datasource-mapping>
```

4. Build the JReport Server WAR file using the makewar.bat utility in the `<install_root>\bin` directory.
5. Deploy JReport Server to JBoss.
6. Run the report that uses this connection.

Configuring connection pool

If you have used SQL or other similar tools to connect to a database and act on data, you probably know that getting the connection and logging in is the part that takes the most time. A program can spend several seconds every time it needs to establish a connection. To improve performance, JReport Server supports caching a connection, so that it can be used each time the same user accesses the same table. Cached connections are kept in the connection pool and can be used and re-used as required. Therefore, JReport Server is freed from having to create the connection to the database, leading to a considerable saving of time.

If you want to improve the performance of JReport Server, you can configure a connection pool using the ConnectionPoolConfig.properties file in the directory `<install_root>\bin`. The following is an example of the ConnectionPoolConfig.properties file and detailed description of each property in the file:

```
# jdbcpool configure information
URL1=jdbc:odbc:Jinfonet
Expire_URL1=0
IdleExpire_URL1=1
MaxCount_URL1=50
MaxShare_URL1=1
Attempt_URL1=1
Interval_URL1=0
```

URL1

JDBC URL that can establish a connection to your database. The URL contents are case sensitive.

This property is a must. You are required to give a value to it in order to establish a connection to the database. The other six properties are optional. If no values are set to them, the default values will be used.

"URL1" is an example of naming the URL. The name should start with the uppercase "URL". If you change the name "URL1" to "URLxY", the "URL1" in the other property names should also be updated to "URLxY", for example, "Expire_URL1" should be changed to "Expire_URLxY".

Expire_URL1

You can define how long in seconds it takes for an active connection to expire. The value for this property by default is "0", which means that the active connection will not expire.

IdleExpire_URL1

You can define how long a free connection can be idle before it is closed. The connection will not be released until the defined time has reached its limit. The value for this property by default is "1", which means that the connection will be released after 1 second. If the value is "0", the connection will be closed right after it starts idling.

MaxCount_URL1

The maximum number of connections based on the URL. The value for this property by default is "50". Once the number of the connections reaches the maximum, a new connection will be blocked until a free connection becomes available. Generally speaking, the fewer the connections, the better the performance of JReport Server. If the value is set to "0", it means that there will be no limit on the number of the connections.

MaxShare_URL1

The maximum number of requests in one connection that can be executed simultaneously. Once the number of the requests reaches the maximum, a new connection will be created to connect to the database. The value for this property by default is "1", which means that one request in one connection can be executed simultaneously. The smaller the number of requests, the better the performance of JReport Server. If the value is set to "0", it means that there will be no limit to the maximum number of requests sharing one connection and that all the requests in one connection can be executed simultaneously.

Since the property MaxShare_URL1 and MaxCount_URL1 are incompatible, you should make a balance between these two properties to achieve the best performance of JReport Server.

Attempt_URL1

This property has a close relation to the property MaxCount_URL1. Once the number of the connections reaches the maximum, a new connection will be blocked until a free connection becomes available. To make the maximum use of the available connections in the pool, the property Attempt_URL1 allows you to set how many times the engine will try to establish a new connection. The default value for this property is "1", which means that the engine will only try once to create a new connection.

Interval_URL1

The interval the engine will wait for between attempts to create a new connection if a previous try has failed. The default value for this property is "0", which means that the engine will start the next attempt immediately. The unit of the property value is milliseconds.

Note: Administrators can access the JReport Administration page > Configuration > Connection Pool panel for viewing connection information and for deleting unused connections.

Configuring logs

JReport provides a robust, flexible and configurable logging system, which is based on log4j version 1.2.8. JReport also supports versions of log4j version 1.2.8 to 1.3alpha7.

The logging system is used for obtaining meaningful and helpful JReport logging information in a convenient way. It is easy to configure and manage. The following are the significant benefits of the JReport logging system:

- It is compatible with your application which is based on log4j.
- It can be configured to output the logging information of different modules separately or to output them all together.
- It supports setting the trace and error type level separately.

Configuring logs

Log configuration tasks can be achieved in three ways. They are:

- Using the JReport Administration page > Configuration > Log panel.
- Using the LogConfig.properties file located in `<install_root>\bin` directory.
- Using command options.

  In comparison with earlier versions, the command options used for log configuration have undergone a considerable change. The following are descriptions of these options:

Options	Description
-vDebug	Enables JReport Engine to output messages to a file and sets engine log file's trace level to INFO and error level to WARN.
-vError	Enables JReport Engine to output messages to a file and sets engine log file's trace level to OFF and error level to ERROR.
-logall	Sets all loggers' trace level to INFO and error level to WARN.
-log[:file Name]	Outputs JReport Engine messages to the file as specified and uses the -vDebug level.

Note: Any settings made using the command options will override the trace and error type level settings in the LogConfig.properties file.

Logging information using the trace and error type

The JReport logging system enables you to trace both normal and abnormal situations by using the trace and error type. Trace is used for logging something expected or regular, such as tracing a program workflow, logging runtime information and associated elements. Error is used for logging something unexpected or irregular. For example, when a URL is unreachable, a file does not exist, or a table cannot be found in a data source.

Trace type is divided into four levels, and is ordered according to the amount of information logged, from the least to the most. These levels are OFF, OUTLINE, INFO, and TRIVIAL. Similarly, error type is also divided into four levels, and is ordered according to the amount of information logged, from the least to the most. These levels are OFF, FATAL, ERROR, and WARN. For detailed information about the eight levels, see Log panel.

If you want the least amount of information to be logged, you can set trace level to OFF and error level to ERROR. If you want the most amount of information to be logged, you can set trace level to TRIVIAL and error level to WARN. Setting these to the highest level could affect system performance as well as disk usage.

Customizing log file location

You can customize where to output logs in your local disk instead of using the default directory `<install_root>\logs`.

By default, different log categories are output to different rolling files. For example, the Engine log is output to Engine.log in the `<install_root>\logs` directory. If you want to customize the location of the Engine log file, take the following steps:

1. Open the LogConfig.properties file located in the `<install_root>\bin` directory.
2. Modify the parameter log4j.appender.EngineRollingFile.File. Set its value to an absolute path with a log file name, for example: `C:\logs\Engine.log` (on Windows) or `/logs/Engine.log` (on Linux/Unix).
3. Save the LogConfig.properties file.
4. Restart JReport Server.

Configuration settings - effect on performance

The following are configuration properties which have an effect on server performance:

- If the showTitle property is set to true in the LogConfig.properties file, the log file size will increase. However, the effect on performance is negligible.
- The levels of trace and error affect the amount of logging output. The higher the level, the more the output. The settings for error level will have a negligible affect on performance but a high level for trace may affect performance.
- The use of the following conversion characters in ConversionPattern property results in the slow generation of caller class or location information: %C, %F, %L and %M. Their use should be avoided unless execution speed is not an issue. For information about the ConversionPattern property, see comments in the LogConfig.properties file, which is located in the `<install_root>\bin` directory.

Changing the server context path

In some circumstances you may want to change the server context path for accessing JReport Server console and running JReport reports. This section introduces how to do this in a standalone or an integrated environment.

Changing standalone server context path

JReport has the following URL configuration for accessing JReport Server console and running JReport reports in a standalone environment:

```
http://<hostname>:8888/jrserver
http://<hostname>:8888/jinfonet/index.jsp
http://<hostname>:8888/dhtmljsp/...
```

You may want to change the server context path as follows:

```
http://<hostname>:8888/jrp/jrserver
http://<hostname>:8888/jrp/jinfonet/index.jsp
http://<hostname>:8888/jrp/dhtmljsp/...
```

To do this:

1. Copy all contents in `<install_root>\public_html` to `<install_root>\public_html\jrp`.
2. Modify the following property values in server.properties in `<install>\bin` directory by adding "/jrp":

   web.design_servlet_path=**/jrp**/webreporting
   web.dhtml_jsp_path=**/jrp**/dhtmljsp
   web.dhtml_servlet_entry_path=**/jrp**/jrdhtml
   web.dhtml_servlet_path=**/jrp**/dhtml
   web.help_servlet_path=**/jrp**/help
   web.jreport_servlet_path=**/jrp**/jrserver
   web.skin.dir=**/jrp**/skin
3. Modify the value of jsp_path in redirect.properties in `<install>\bin` directory by adding "/jrp":

   jsp_path=**/jrp**/jinfonet/
4. Modify mapping.properties in `<install>\bin` directory by adding "/jrp":

   ```
   # Map servlets to paths
   # Properties beginning with a . are extension properties, all other
   # properties are path properties
   #
   # Format:
   # path or extension = servlet name
   ```

   /jrp/jrserver=jrserver
   /servlet/sendfile=sendfile
   /jrp/dhtml=dhtml
   /jrp/jrdhtml=jrdhtml
   /jrp/help=help
   .jsp=jspservlet
5. Start the server and access the server by the URLs:

   http://localhost:8888/jrp/jrserver
   http://localhost:8888/jrp/jinfonet/index.jsp

   Run a report in Page Report Studio by the URL:

   http://localhost:8888/jrp/dhtmljsp/dhtml.jsp?jrs.catalog=%2fSampleReports%2fSampleReports.cat&jrs.report=%2fSampleReports%2fEmployee Information.cls

Changing integrated server context path

Supposing that all server resources are deployed to the context root folder jreport\ after deploying JReport Server to an application server.

You can use the following URLs for accessing JReport Server service console and running JReport reports in the integrated environment:

```
http://<hostname>:port/jreport/jinfonet/index.jsp
http://<hostname>:port/jreport/dhtmljsp/...
```

You may want to change the server context path as follows:

```
http://<hostname>:port/jreport/myjsp/jinfonet/index.jsp
http://<hostname>:port/jreport/myjsp/dhtmljsp/...
```

To do this:

1. Move index.htm and the following folders from `jreport\` to `jreport\myjsp\`:

   admin
   dhtmljsp
   images
   javascript
   jinfonet
   skin
   style
2. Create a file jrserver.properties in the \WEB-INF directory, add the skin and dhtmljsp properties and provide the correct paths (the context root is excluded):

   web.skin.dir=/myjsp/skin
   web.dhtml_jsp_path=/myjsp/dhtmljsp
3. Access the server by the URL: `http://<hostname>:port/jreport/myjsp/jinfonet/index.jsp`

   Run a report in Page Report Studio> by the URL:

   http://<hostname>:port/jreport/myjsp/dhtmljsp/dhtml.jsp?jrs.catalog=%2fSampleReports%2fSampleReports.cat&jrs.report=%2fSampleReports%2fEmployee Information.cls

Managing JReport Server

The administration task in JReport Server consists of six major parts. They are management of resources, versions, security, tasks, server data and cached report data. This chapter discusses these task types in detail. After going through these topics, you will be able to manage your JReport Server more effectively.

- Managing resources
- Managing versions
- Managing security
- Managing tasks
- Managing server data
- Managing cached report data
- Managing in-memory cubes

Note: A JReport Live license for JReport Server is required in order to use web report related features, and a JDashboard license is required in order to use dashboard and library component related features. If you do not have the licenses, contact your Jinfonet Software account manager to obtain them.

Managing resources

Resources on JReport Server are organized into a tree structure called the Resource Tree. Only the resources that are organized in the resource tree can be accessed and queried by a client.

You can perform basic resource tasks, such as publishing, deleting, getting resources from a real path, and more advanced tasks, such as changing resource and folder properties. Moreover, you can also manage the versions and secure your resources. Detailed information is available in sections - Managing versions and Managing security.

Note: Catalogs are by default not displayed in the server resource tree on the JReport Console page. In order to perform operations on catalogs published from outside of JReport Server on the JReport Console page, you need to enable them to be displayed first by setting the web.page.option. show_catalog property to true in the server.properties file.

This section includes the following topics:

- Publishing resources
- Converting resources of earlier versions into current version
- Getting and using resources from a real path
- Customizing TTF font location for resources
- Working with custom fields
- Changing resource and folder properties
- Linking resources and catalogs
- Managing dynamic display names of business view elements
- Deleting resources

Publishing resources

Before you can perform any tasks on JReport Server, you first need to have your resources published and organized. The resources are reports, library components, catalogs, and folders.

The JReport Server resource tree contains four built-in folders: My Components and Public Components are used to store library components with catalogs, while My Reports and Public Reports store reports with catalogs.

There are three ways of publishing resources:

- Publishing resources from JReport Designer
- Publishing resources from a local computer
- Publishing resources from a remote computer

Note: After resources have been delivered to JReport Server, you may find that the publish time shown on the server UI is different from the client time. That is because the publish time takes the server time instead of the client time.

Publishing resources from JReport Designer

You can directly publish your reports, library components, or catalogs from JReport Designer to JReport Server.

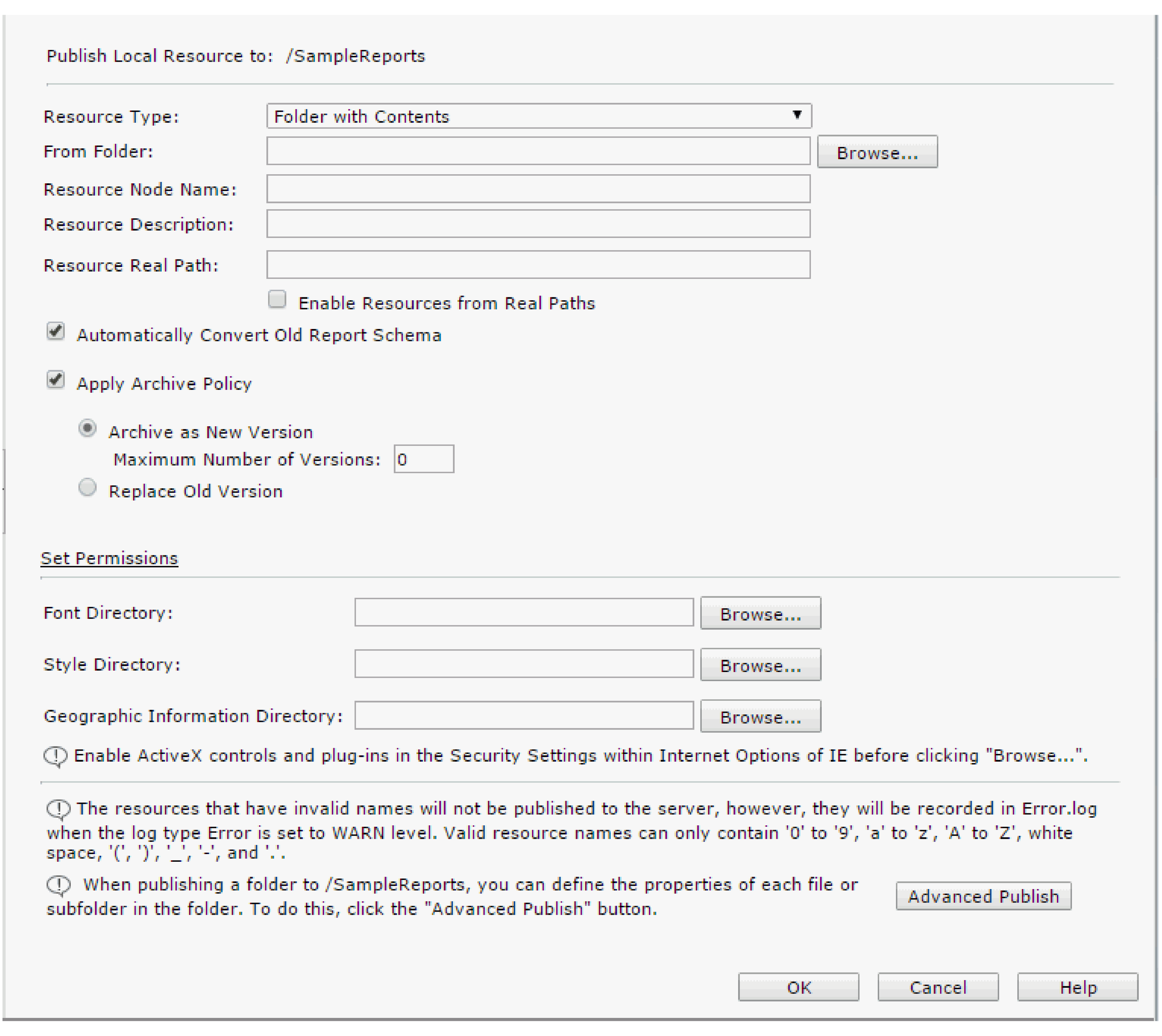
Publishing resources from a local computer

You can publish these types of resources from a local computer. They are reports, library components, catalogs, and folders.

To publish resources from a local computer:

1. Do either of the following:
   - For administrators, on the JReport Administration page, click **Resources** on the system toolbar and select **Resources** from the drop-down menu. In the Resources panel, browse to the folder where you want to publish the resources, then click the **Publish to Local Server** link.
   - For end users, on the JReport Console > Resources page, browse to the folder in which to publish the resources, then on the task bar of the Resources page, click **Publish** > **To Local Server**.

   The Publish to Local Server dialog appears.

2. Select a type from the Resource Type drop-down list.
3. Specify properties for the to-be-published resources as required.
   - **If the Page Report or Web Report resource type is selected:**
     1. In the From File text field, specify the report you want to publish with its full path.
     2. In the Resource Node Name field, specify a name for the report. This name is required and is used as the display name of the report in the server resource tree.
     3. Type a brief description to describe the report in the Resource Description field.
     4. From the Status drop-down list, select the status of the report.
     5. Specify values of the custom fields for the report if there are custom fields.
     6. Click the **Browse** button to specify a font directory for the report.
     7. Click the **Browse** button to specify a style directory for the report.
     8. Click the **Browse** button to specify a geographic information directory for the report.
   - **If the Catalog resource type is selected:**
     1. In the From File text field, specify the catalog you want to publish with its full path.
     2. In the Resource Node Name field, specify a name for the catalog. This name is required and is used as the display name of the catalog in the server resource tree.
     3. Type a brief description to describe the catalog in the Resource Description field.
     4. Specify values of the custom fields for the catalog if there are custom fields.
   - **If the Component resource type is selected,**
     1. In the From File text field, specify the library component you want to publish with its full path.
     2. In the Resource Node Name field, specify a name for the library component. This name is required and is used as the display name of the library component in the server resource tree.
     3. Type a brief description to describe the library component in the Resource Description field.
     4. Specify values of the custom fields for the library component if there are custom fields.
     5. Click the **Browse** button to specify a font directory for the library component.
     6. Click the **Browse** button to specify a style directory for the library component.
     7. Click the **Browse** button to specify a geographic information directory for the library component.
   - **If the Folder resource type is selected:**
     1. In the Resource Node Name field, specify a name for the folder. This name is required and is used as the display name of the folder in the server resource tree.
     2. Type a brief description to describe the folder in the Resource Description field.
     3. In the Resource Real Path text field, type a local disk path as the real path of the folder. If you want to enable getting resources from the real path, check **Enable Resources from Real Paths**. The local disk resources will be mapped into the resource node of the folder in the server resource tree and the server will be able to always get resources and updates from the real path. For details, see Getting and using resources from a real path.

        If this step is skipped, the publish job will simply create a new blank directory node in the server resource tree.
     4. Specify values of the custom fields for the folder if there are custom fields.
   - **If the Folder with Contents resource type is selected:**
     1. Click the **Browse** button next to the From Folder text field to specify the folder where the resources you want to publish are saved.
     2. In the Resource Node Name field, specify a name for the folder. This name is required and is used as the display name of the folder in the server resource tree.
     3. Type a brief description to describe the folder in the Resource Description field.
     4. In the Resource Real Path text field, type a local disk path as the real path of the folder. If you want to enable getting resources from the real path, check **Enable Resources from Real Paths**. The local disk resources will be mapped into the resource node of the folder in the server resource tree and the server will be able to always get resources and updates from the real path.
     5. Specify values of the custom fields for the folder if there are custom fields.
     6. Click the **Browse** button to specify a font directory for the resources.
     7. Click the **Browse** button to specify a style directory for the resources.
     8. Click the **Browse** button to specify a geographic information directory for the resources.
     9. If you want to use Advanced Publish, click the **Advanced Publish** button. All resources contained within the specified folder will then be displayed. Check the resources you want to publish and specify the properties for each resource as required.
   - **If the Catalogs, Reports and Folders in Folder or Catalogs and Reports in Folder resource type is selected:**
     1. Click the **Browse** button next to the From Folder text field to specify the folder where the resources you want to publish are saved.
     2. Click the **Browse** button to specify a font directory for the resources.
     3. Click the **Browse** button to specify a style directory for the resources.
     4. Click the **Browse** button to specify a geographic information directory for the resources.
     5. If you want to use Advanced Publish, click the **Advanced Publish** button. All resources contained within the specified folder will then be displayed. Check the resources you want to publish and specify the properties for each resource as required.
4. If the resources you specify to publish contain reports created in earlier versions, check **Automatically Convert Old Report Schemas**, the reports will then be automatically converted into current version JReport reports when publishing finishes.
5. To apply an archive policy to the resources that you are publishing,
   - For resources published to the My Reports or Public Reports folder, check the **Apply Archive Policy** option, then specify the archive policy as required: Archive as New Version or Replace Old Version.
   - For resources published to the My Components or Public Components folder, specify the maximum number of versions that will be listed in the version table of the resources.

   Note that a folder by itself does not have versions; the archive policy specified for a folder applies to the folder content.
6. If the resources are to be published to the Public Reports or Public Components folder, click the **Set Permissions** link to specify user permissions to them according to your requirement.
7. Click **OK** to start publishing the resources.

Note: The Publish to Local Server option is always available to administrators on the JReport Console page. For end users, the option is displayed only when they have the privilege of publishing resources and have logged onto JReport Server from a local browser.

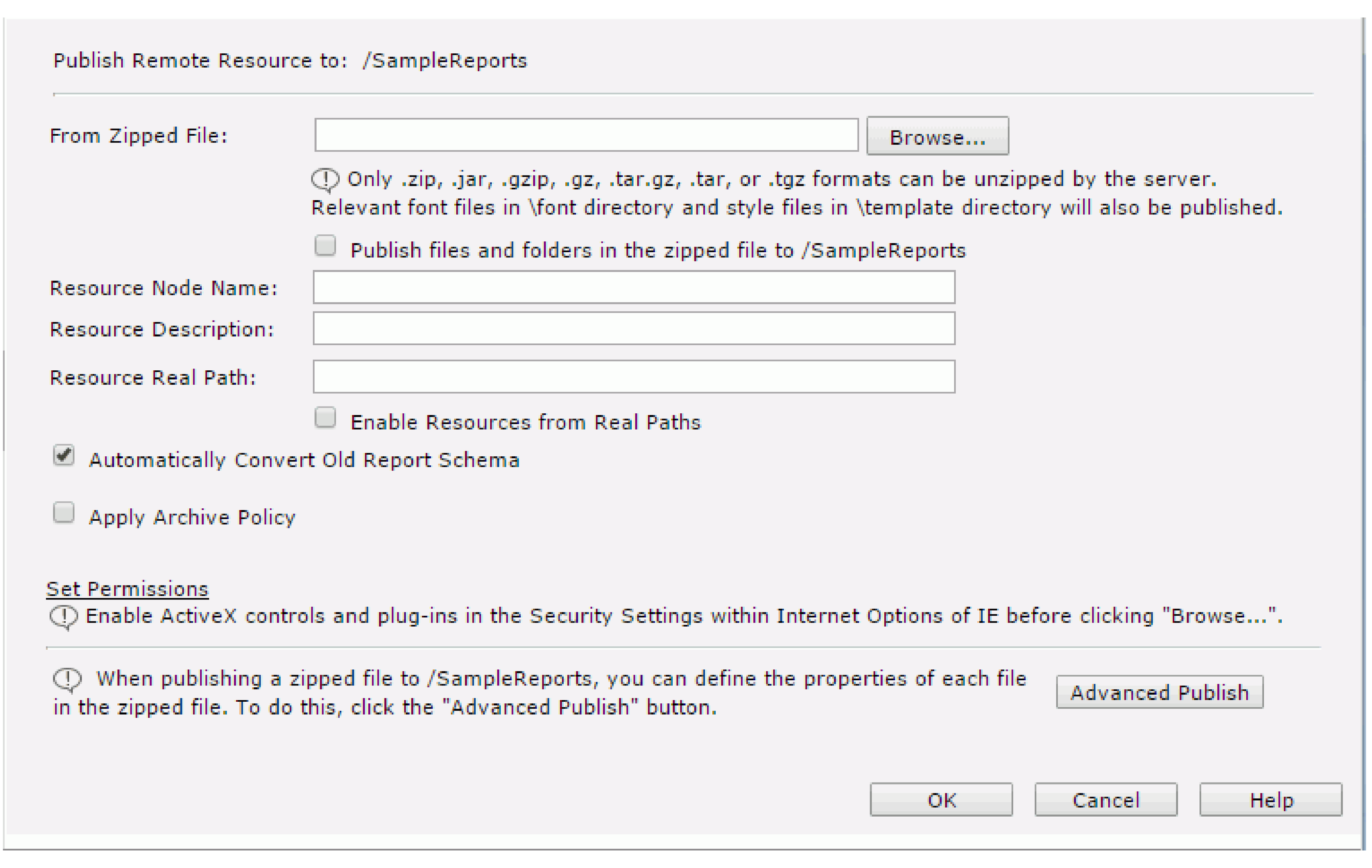
Publishing resources from a remote computer

You can publish these types of resources from a remote machine to the server. They are reports, library components, catalogs, and folders. Only administrators can publish library components. Before publishing remote resources, you must create a zip file or tar file containing the resources to publish.

To publish resources from a remote computer:

1. Do either of the following:
   - For administrators, on the JReport Administration page, click **Resources** on the system toolbar and select **Resources** from the drop-down menu. In the Resources panel, browse to the folder where you want to publish the resources, then click **Publish to Remote Server**.
   - For end users, on the JReport Console > Resources page, browse to the folder in which to publish the resources, then on the task bar of the Resources page, click **Publish** > **To Remote Server**.

   The Publish to Remote Server dialog appears.

2. Click the **Browse** button to specify the zipped file which contains the resources you want to publish.
3. Specify where the resources will be published.
   - If you want to publish the resources directly to the current open folder, check the **Publish files and folders in the zipped file to /XXX** checkbox.
   - If you want to create a new folder in the current open folder to locate the resources,
     a. Make sure the Publish files and folders in the zipped file to /XXX checkbox is not selected.
     b. In the Resource Node Name box, specify the name of the folder. The name will then be used as the display name of the folder in the server resource tree.
     c. Type a brief description to describe the folder in the Resource Description box.
     d. Specify values of the custom fields for the folder if there are custom fields.
     e. In the Resource Real Path text field, type a local disk path as the real path of the folder. If you want to enable getting resources from the real path, check **Enable Resources from Real Paths**. The local disk resources will be mapped into the resource node of the folder in the server resource tree and the server will be able to always get resources and updates from the real path. For details, see Getting and using resources from a real path.
4. If the resources you specify to publish contain reports created in earlier versions, check **Automatically Convert Old Report Schemas**, the reports will then be automatically converted into current version JReport reports when publishing finishes.
5. To apply an archive policy to the resources that you are publishing,
   - For resources published to the My Reports or Public Reports folder, check the **Apply Archive Policy** option, then specify the archive policy as required: Archive as New Version or Replace Old Version.
   - For resources published to the My Components or Public Components folder, specify the maximum number of versions that will be listed in the version table of the resources.
6. If the resources are to be published to the Public Reports or Public Components folder, click the **Set Permissions** link to specify user permissions to them according to your requirement.
7. If you want to use Advanced Publish, click the **Advanced Publish** button. All resources contained within the zip file will then be displayed. Check the resources you want to publish and specify the properties for each resource as required.
8. Click **OK** to start publishing the resources.

Note: When publishing resources from a remote computer, the process is similar to that for a local publish. However, there are differences:

- Local Publish publishes resources from the machine where the server runs, while a Remote Publish publishes resources from a client machine using a web browser to the machine where the server runs.
- The resource type of Remote Publish can ONLY be a compressed file. You should compress the reports, library components, and catalog files in advance, and if special fonts are used in the reports or library components, add the corresponding font files into the compressed file as well. There are two approaches to building a compressed file.
  - You can compress the resources manually using a third-party tool, such as Winzip and gzip.
  - You can use jar.exe that the JSDK provides to build a compressed jar file directly. Use the command as follows:

    ```
    %JAVAHOME%\bin\jar.exe -cvfM %DEST_JAR_FILE% %SOURCE_RESOURCES%
    ```

Parameter	Description
%JAVAHOME%	The Java SDK install root.
%DEST_JAR_FILE%	The destination file path and file name. The .jar file will be generated to the path you specify here, using the file name you provide.
%SOURCE_RESOURCES%	The source file path and file name. Note that specifying a path for this parameter will cause the generated jar file to contain the same path information. For example, when you extract a jar file compressed using `myReports\*.*` for this parameter, the files will be extracted to a folder called `myReports`. JReport Server is not able to import a compressed file which contains the path information, so do not specify a path for this parameter.

    To generate a jar file containing no path information, switch to the source folder, and then carry out the compression. For example,

    ```
    C:\myReports>C:\jdk1.6.0_17\bin\jar -cvfM c:\temp\aa.jar.
    ```

    The jar file will be generated to `c:\temp`, as aa.jar, compressing all the files in `c:\myReports`, and containing no path information.

    Always use this method if the folder you are compressing contains reports or library components with Chinese, Korean, or Japanese names.

Converting resources of earlier versions into current version

If your JReport Server contains resources earlier than V8 such as reports, visual analysis, library components, dashboards, catalogs, etc., during each server runtime lifecycle when you run such a resource for the first time, the JReport Engine has to first convert the earlier version resource into a current version resource before running the resource. This may result in decreased performance. To avoid this, JReport Server allows administrators to permanently convert resources earlier than V8 in the server resource tree to adapt to the current version.

Converting via the Administration UI

The converting UI is available to administrators only. It is used to convert all resources earlier than V8 in the My Reports folder or Public Reports folder. The converted resources are stored in the same directory.

To convert resources earlier than V8 into current version:

1. On the JReport Administration page, click **Resources** on the system toolbar, then select **Converting** from the drop-down menu.
2. Specify the folders which contain the resources you want to convert.
3. Specify how to save the converted resources.
   - **Replace the Old Resource**
     Replaces the old resources with the converted resources.
   - **Save as New Versions**
     Saves the converted resources as new resource versions in the resource tree.
4. When done, click **OK** to start converting.

Converting using tool

You can also use rptconv.bat/rptconv.sh located in the `<install_root>\bin` directory to convert a resource and a type of resources or all resources earlier than V8 in a directory to current version.

Getting and using resources from a real path

Not only can you use the published resources described in section Publishing resources, but also those resources obtained from a real path. The difference between real path resources and published resources lies in the way of managing them. Published resources are managed in JReport Server totally. Real path resources are managed in both the server and the operating system. In the server, when accessing a resource node which is linked with a real path, the local resources in the real path are loaded to the node and displayed together with other server resources (including the published resources) in the node. You cannot delete the real path resources directly from the server UI, but can only achieve it by removing the resources from the local disk.

To use real path resources, you should specify a real path for a virtual folder. You can then add or remove resources into or from JReport Server by adding or removing them into or from the real path. Also, if you do not want to publish resources to the server by using the publish feature, you can set a real path instead.

The following example uses the resources in the folder Public Reports obtained from the real path:

1. On the JReport Administration page, click **Configuration** on the system toolbar, then select **Advanced** from the drop-down menu.
2. In the Advanced panel, check the **Enable Resources from Real Paths** option, save the changes and then restart the server.
3. On the JReport Console page, open the Properties dialog of the folder /Public Reports by putting the mouse pointer over the folder row and clicking the Properties button on the floating toolbar, check **Enable Resources from Real Paths**, specify a real path for the demo virtual folder as `C:\JReport\Server\reportbak`, then click **OK** to save the changes.
4. In the Windows directory `C:\JReport\Server\reportbak`, create three sub directories report1, report2, and report3, and then copy some reports and catalogs into them.
5. On starting JReport Server, click the virtual folder **/Public Reports**. You will then see the three sub folders: report1, report2, and report3 within it, including their reports and catalogs.

Now end users can perform operations on the real path resources, including viewing versions, setting properties, and report based operations such as running, advanced running, and scheduling.

Notes:

- You cannot publish resources to a folder that comes from a real path.
- You cannot use a real path resource as an archive location.
- Real path resources themselves can only have one version in the server resource tree, so they do not support the Archive Policy feature for managing multiple versions.
- The relationships of folder real paths do not need to be consistent with that of the resource tree folders. That is, they are independent of each other. For example, in the resource tree, the folder ABC, whose real path is `C:\FolderABC`, has a sub folder DEF. The real path of DEF need not necessarily be a folder in `C:\FolderABC`. It can be any folder.
- After using the real path resources, under certain special conditions some invalid nodes will be generated in the server databases. In order to delete the invalid nodes, administrators can perform the corresponding option on the JReport Administration page to clear unused nodes from the internal JReport Server database. For details, see Clearing unused nodes from the system database.

Setting default real paths for the Public Reports and Public Components folders before server starts

JReport Server enables setting the default real paths for the Public Reports and Public Components folders without having to log onto the server. There are two ways to achieve this goal: setting the real path during or after installation.

- **Setting during installation**

  1. When installing JReport Server using the Installation Wizard, select the installation type **Custom Installation for Standalone Server**. The Configure System Environment screen is then available.
  2. In the screen, select **Advanced** from the left panel.
  3. Check the **Enable Resources from Real Paths** option, and then type in the paths separately in the Public Reports Real Path and Public Components Real Path text fields or specify the paths using the Browse button.
  4. Upon completion of the installation, an install.server.properties file will be created in the `<install_root>\bin` directory. When the server starts, it retrieves the file contents, and then sets the real path for the Public Reports and Public Components folders. Note that the file is deleted after the server has loaded the file contents.

- **Setting after installation**

  If you want to set the default real paths for the Public Reports and Public Components folders after installation, you can create an install.server.properties file in the `<install_root>\bin` directory manually. Specify the content in the file as follows:

```
server.publicReportsRealPath=D:\JReport
server.publicComponentsRealPath=D:\JReport
```

Customizing TTF font location for resources

The default font path in JReport Server is `<install_root>\font`. You can set the font path to a different location by using the following three ways. These are in an order from higher to lower priority:

- Use the -D parameter to set the system properties key jreport.server.font.path
- Use the API HttpUtil.initEnv() to set the properties key server.font.path
- Add a property server.font.path in the server.properties file and set a full path to it.

Notes:

- The value specified by -D is not stored into the server.properties file.
- The value specified using API is stored into the server.properties file.

Working with custom fields

Custom fields are user defined fields which can be used as resource properties, the same as Type, Description, Last Modified, and so on.

After a custom field is created and enabled by administrators, it will be available in the resource information table on both the JReport Administration > Resources panel and the JReport Console > Resources page. End users can define its value by setting properties.

Creating and enabling custom fields

To create a custom field:

1. On the JReport Administration page, click **Configuration** on the system toolbar, then select **Custom Field** from the drop-down menu.
2. In the Custom Field panel, click **New Custom Field** on the task bar.
3. In the New Custom Field dialog, provide a name and description for the custom field.
4. To enable the custom field, check the **Enabled** option.
5. Click **OK** to create the custom field.

Editing custom fields

To edit a custom field:

1. In the Custom Field panel, select the row where the custom field is in, then click **Edit** > **Properties** on the task bar.
2. In the displayed dialog, modify the name and description for the custom field, and change its status if required.
3. Click **OK** to accept the changes.

Deleting custom fields

Administrators can delete the custom fields that are not required at any time. To do this:

1. In the Custom Field panel, select the row where the custom field is. You can select multiple custom fields.
2. Click **Edit** > **Delete** on the task bar.
3. Click **OK** in the warning message to confirm the deletion.

Setting value to custom fields

When you publish resources to JReport Server, either locally/remotely or from JReport Designer, if there are custom fields enabled, they will be displayed in the publishing dialog and you can specify the value of the custom fields for each resource according to your requirements. Also, the custom field values can be defined by setting resource properties.

Hiding custom fields

By default, all the enabled custom fields will be displayed in the resource information table on both the JReport Administration > Resources panel and the JReport Console > Resources page. If you want to hide a custom field from being shown in this table, follow the steps below:

For administrators:

1. Click **Profile** on the system toolbar of the JReport Administration page, then select **Customize Server Preferences** from the drop-down menu.
2. In the General tab, in the Columns Shown in Resources List section, unselect the checkbox in front of the corresponding custom field.
3. Click **OK** to accept the setting.

For end users:

End users can hide a custom field on the JReport Console page > Profile > Customize Server Preferences > General tab as shown in the above method, and besides have a second choice:

1. On the JReport Console > Resources page, click **Tools** > **Preferences** on the task bar.
2. In the General tab of the Preferences dialog, unselect the checkbox in front of the corresponding custom field, then click **OK**.

Changing resource and folder properties

To change the properties of a resource or folder:

- For administrators, on the JReport Administration page, click **Resources** on the system toolbar and select **Resources** from the drop-down menu. In the Resources panel, browse to the row that the resource or folder is in and click the **Properties** button in the Control column. In the corresponding properties dialog, change the settings as required.
- For end users, on the JReport Console > Resources page, browse to the row that the resource or folder is in and do either of the following:
  - Select the row and click **Tools** > **Properties** on the task bar of the Resources page.
  - Select the row, right-click in the row and select **Properties** from the shortcut menu.
  - Put the mouse pointer over the row and click the **Properties** button on the floating toolbar.

  Then, in the corresponding properties dialog, change the settings as required.

See also the following documents for details about the property settings:

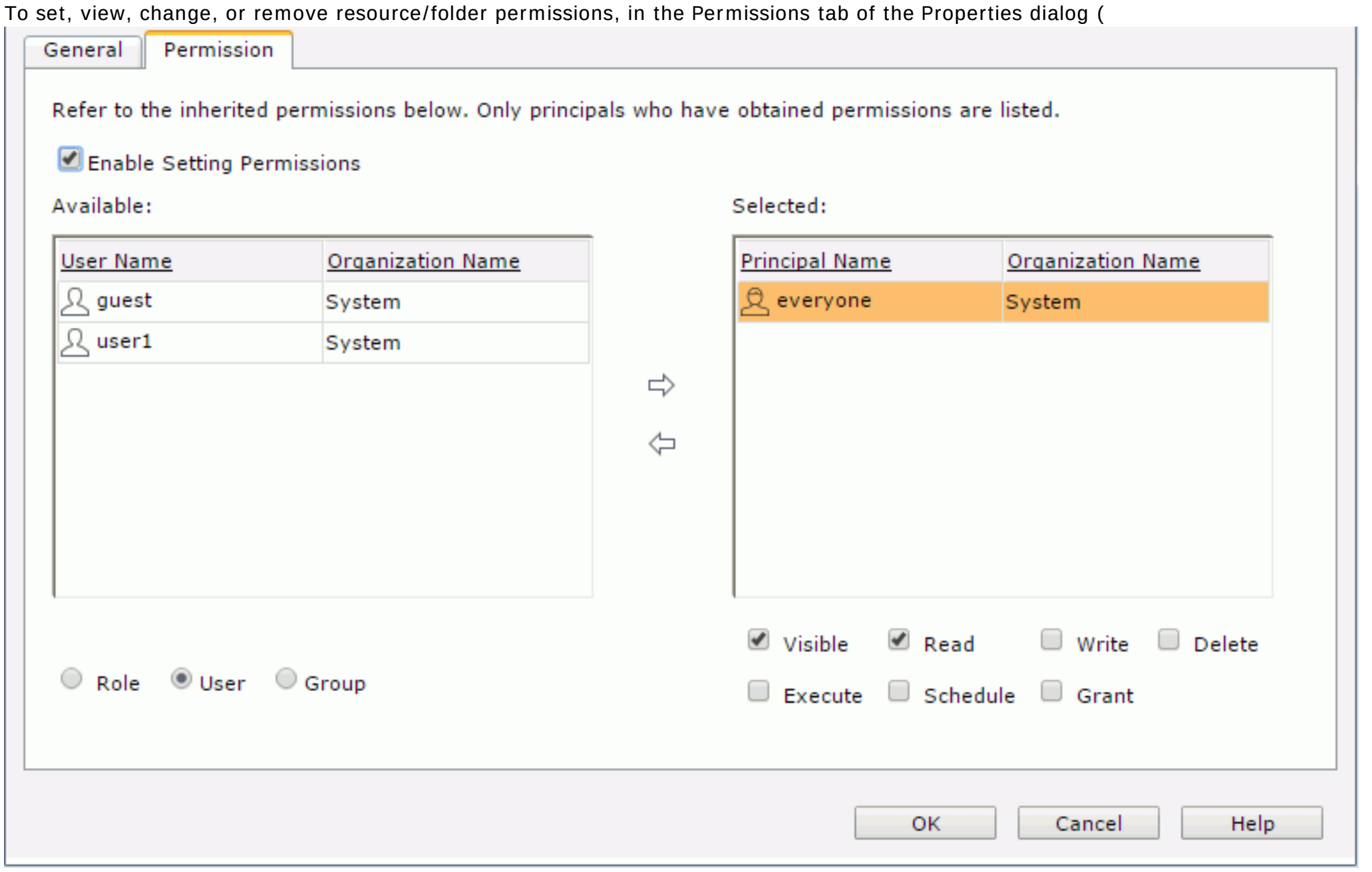
- Catalog Properties dialog
- Dashboard Properties dialog
- Folder Properties dialog
- Library Component Properties dialog
- Report Properties dialog
- Result Properties dialog
- Visual Analysis Template Properties dialog

Assigning permissions

Permissions, associated with resources and folders which locate in the Public Reports or Public Components folder, are the rules that are granted to users to control their access to resources and folders.

After you have set permissions for a parent folder, any new resources and sub folders created in that folder will inherit the same permissions. If you do not want them to inherit these permissions, you can enable their user permissions and set their permissions separately. Resources and folders will inherit permissions from their parent folder if their user permissions are not enabled.

To set, view, change, or remove resource/folder permissions, in the Permissions tab of the Properties dialog (



To set up or change permissions for a role, user or group, first check **Enable Setting Permissions**, then select the role/user/group in the Selected box and check or uncheck the required permissions. If the role/user/group is not listed in the Selescted box, select the corresponding radio button below the Available box, add the role/user/group to the Selected box and then assign the permissions accordingly.

- To remove resource/folder permissions for all users, groups and roles, uncheck the **Enable Setting Permissions** option.

Notes:

- Security permissions do not apply to the built-in version folders, the My Reports and My Components folders, and their contents.
- To complete a task, you may require more than one permission. For example, to view the properties of a report, you must have both the Visible and Read permissions.
- Some permissions depend on other permissions in order to work, such as Write, Execute, and Schedule. Allowing anyone of these will also allow the Read permission.
- Only members in the administrator role can assign the Grant permission to other users or groups or roles.
- Users that are given the Grant permission can grant permissions to other users in the same group.

Linking resources and catalogs

A resource such as a report or a library component can be linked with a catalog in JReport. The benefits of a linked catalog compared to a copied catalog are:

- There is no need to also copy the wanted catalog to the destination directory when saving the resource to a different location.
- When the resource and its linked catalog are not in the same directory, the resource can still run with the catalog.
- When the linked catalog is updated, the resource using the catalog can run with the updated version. However, the copied catalog cannot be updated if its original catalog is updated since they are two independent versions.

When directly running a resource, the linked catalog has higher priority than the catalog specified in the same folder as the resource (without linked catalog, the resource will run within the selected catalog in the same folder). As for Advanced Run and Schedule, the default selected catalog is the linked catalog if there is one, however, you can change it by the provided option Select Another Catalog.

Setting linked catalog

Linked catalog can be set at server level, folder level, and resource level as follows:

- To set linked catalog at server level (this can only be done by administrators):
  1. Go to the JReport Administration page, click **Configuration** on the system toolbar and select **Advanced** from the drop-down menu.
  2. In the Advanced panel, check **Enable Linked Catalog**, then click the **Select Another Catalog** link to specify the catalog which will be used as the linked catalog at server level.
  3. Click **Save** to save the change.
  4. Restart JReport Server to make the settings take effect.
- To set linked catalog at folder/resource level, go to the Properties dialog of the folder/resource, check **Enable Linked Catalog**, then specify the linked catalog as required.
  - **Use Specified** - If checked, you can specify a linked catalog which can be any catalog in the server resource tree to the folder/resource.
  - **Use Inherited** - If a linked catalog has been specified on the parent level of the folder/resource, you can use the parent-level linked catalog as the linked catalog of the folder/resource. For the My Components, Public Components, My Reports, and Public Reports folders, the parent level is the server level; for the other folders or resources, the parent level is the parent folder.

Notes:

- If a schedule task has been submitted and then the linked catalog in use is modified, the task will still use the previous catalog until the task information is updated.
- When saving a page report in Page Report Studio, you can also save the original catalog as a linked catalog for the page report. It is the same case when saving a web report in Web Report Studio.
- The saved Visual Analysis templates are linked to their original catalogs. However, currently Visual Analysis templates themselves cannot be customized to link to a different catalog.

Managing dynamic display names of business view elements

Administrators can define dynamic business view element display names for different SIDs (security identifiers that refer to the users, roles and groups in the JReport Server security system). Then, when any user of a specified SID logs onto JReport Server, the dynamic display names defined for the SID will be shown in the business view resource tree.

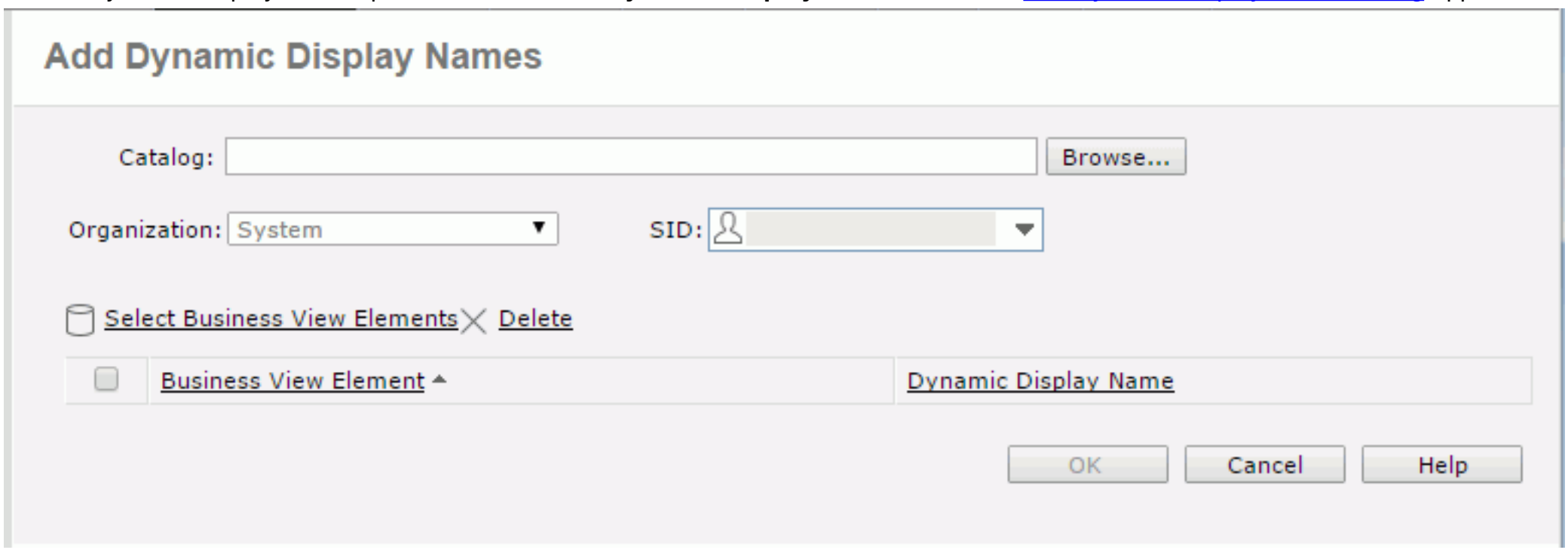
Dynamic display name resources have higher priority than NLS resources.

For business view elements that are not given dynamic display names, their original display names will be used.

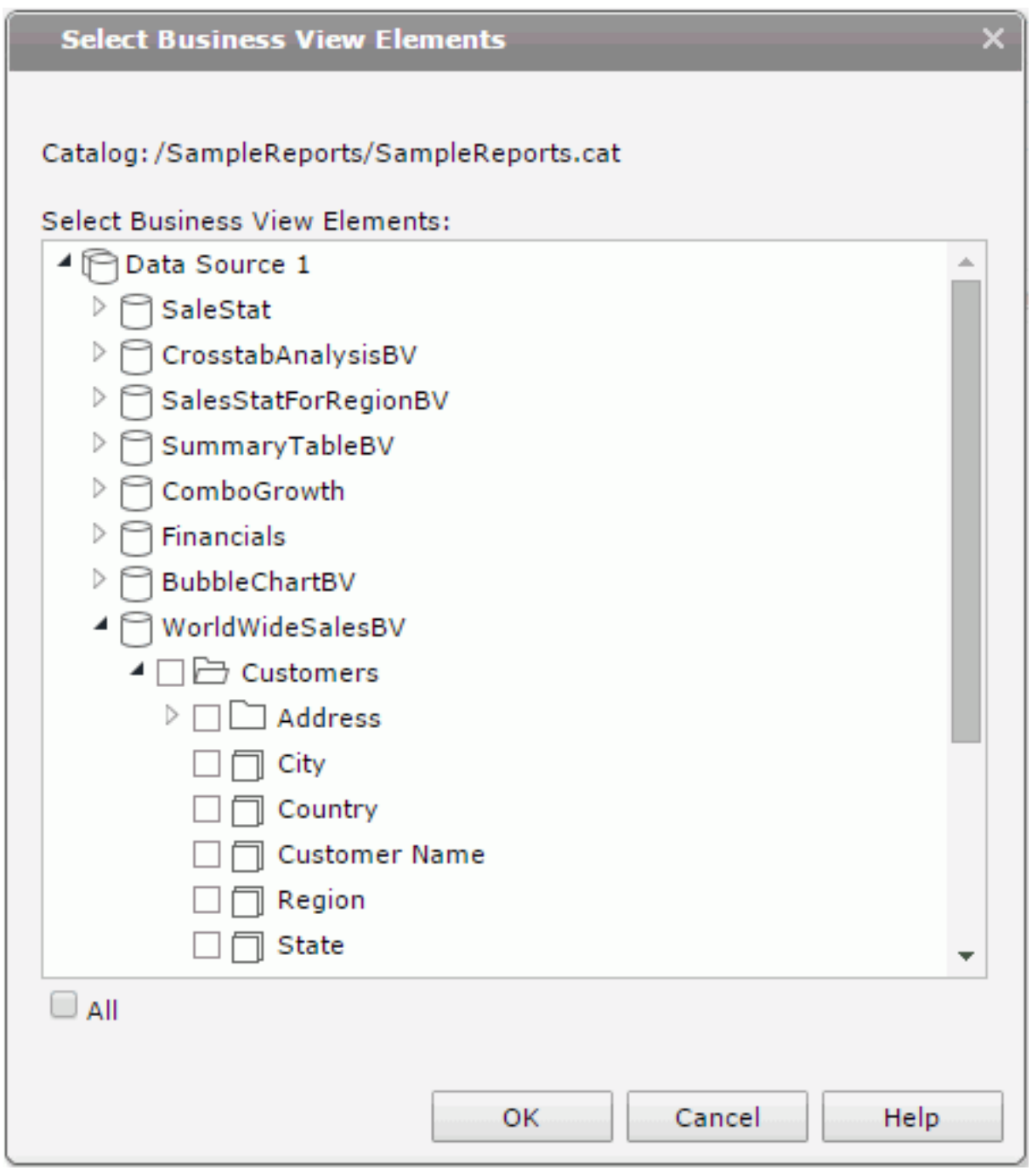
Adding dynamic display names for business view elements

To create dynamic display names for business view elements in a catalog, take the following steps:

1. On the JReport Administration page, select **Configuration** > **Dynamic Display Names** on the system toolbar.
2. On the Dynamic Display Names panel, click the **Add Dynamic Display Names** link. The Add Dynamic Display Names dialog appears.



3. In the Catalog text box, type in the catalog resource path, or click **Browse** to select one from the server resource tree.
4. If the Organization option is available and editable, select an organization from the drop-down list. System is a category to include non-organization users.
5. From the SID drop-down list, select a user, role or group for whom the dynamic display names will take effect. If left blank, it means all the users (when the organization feature is enabled, blank means all the users in the specified organization).
6. Click the **Select Business View Elements** link, and the Select Business View Elements dialog appears.



7. Select the business view elements in the catalog that you would like to define dynamic display names for the specified SID. To select all of the elements, select the **All** checkbox. Then click **OK** to go back to the Add Dynamic Display Names dialog.
8. The selected business view elements will be listed in the business view elements table. Change their display names in the Dynamic Display Name column: double-click in a name text box to enter the editing mode, change the name, then click outside of the text box to accept the change.

   To remove a view element from the table, check the checkbox ahead of it, then click the **Delete** link above the table. You can select multiple elements and remove them at a time.
9. Click **OK** to finish editing the names.

   A new dynamic display name resource row is now added on the Dynamic Display Names panel.

Editing dynamic display names

To edit the dynamic display names defined for a specific SID, on the Dynamic Display Names panel, click its corresponding **Edit** link in the Dynamic Display Names column. In the Edit Dynamic Display Names dialog, edit the dynamic display names as required. You can make the dynamic display names applied to another SID if you want.

Importing/exporting dynamic display names from/to a resource bundle file

A resource bundle file is a properties file with .properties as the suffix. It defines dynamic display names by key/value pairs. The key is a qualified object name. The format is *[Data Source Name]***.***[Business View Name]***.***[Category Name]***.***[Element Name]*, for example, Data Source 1.WorldWideSalesBV.Customers.Customer Name. The value is a dynamic display name.

Here is an example of the contents of a resource bundle file:

```
Data Source 1.SalesStatForRegionBV.Account Managers.Assigned Region=Region
Data Source 1.WorldWideSalesBV.Customers.Customer Name=Customer
Data Source 1.SalesStatForRegionBV.Account Managers=Managers
```

To import dynamic display names from a resource bundle file:

Locate the dynamic display name resource defined for a specific SID on the Dynamic Display Names panel and click **Import** in the Dynamic Display Names column. In the Select Dynamic Display Name File dialog, click **Browse** to select a local resource bundle file that defines the dynamic display names of the business view elements in a catalog and click **OK**. The dynamic display names defined in the resource bundle file will overwrite those in the current dynamic display name resource.

To export dynamic display names to a resource bundle file:

Locate the dynamic display name resource defined for a specific SID on the Dynamic Display Names panel and click **Export** in the Dynamic Display Names column. The dynamic display names defined in the resource will be exported to a resource bundle file.

Deleting resources

To delete a resource or folder:

- For administrators, on the JReport Administration page, browse to the row that the resource or folder is in, then click the **Delete** button in the Control column.
- For end users, on the JReport Console > Resources page, browse to the row that the resource or folder is in, then do either of the following:
  - Select the row, right-click in the row and select **Delete** from the shortcut menu.
  - Put the mouse pointer over the row and click the **Delete** button on the floating toolbar.

Notes:

- The Public Components, My Components, Public Reports and My Reports folders cannot be deleted.
- Deleting a resource or folder removes the resource or folder from disk. The deleted resources and folders cannot be retrieved. Any relevant versions belonging to the resource will also be deleted.

Managing versions

Your resources might change over time. JReport Server uses a versioning system to create and manage resources that have changed in content and properties owing to updates made to them.

All the resources in the server resource tree: reports, report results, dashboards, library components, and catalogs, are controlled by versions. A great proportion of resource management tasks are carried out by managing resource versions.

In addition, JReport Server uses an archive policy to control the resource versions. You can control whether or not to use multiple versions for a specific resource. Also, you can define the maximum number of versions that can be listed in the version table. The archive policy can be applied to a single resource individually, or to many resources in a folder as a whole.

This section shows how to manage resource versions as follows:

- Creating versions
- Browsing versions
- Applying an archive policy
- Deleting versions
- Advanced version topics

Creating versions

Generally, report result versions are created when an end user runs a report using Advanced Run or Schedule Run. However, to create a new version for a report, library component, or catalog, you have to publish them respectively from outside of JReport Server, and then make the old and the new share one common resource node. Versions of a dashboard are generated by saving the dashboard, and each saving action will create a version in the dashboard.

Creating report, library component, or catalog versions

The method for creating a new version to a report, library component, or catalog is by publishing a resource of the same type with exactly the same name to the same location on JReport Server. For how to publish, see Publishing resources.

When publishing resources, you need to apply an archive policy so as to make the published resources saved into the existing resources as a new version.

Note: Catalogs are by default not displayed in the server resource tree on the JReport Console page. In order to perform operations on catalogs published from outside of JReport Server on the JReport Console page, you need to enable them to be displayed first by setting the web.page.option. show_catalog property to true in the server.properties file.

Creating report result versions

A report result version can only be created by advanced running the report or by scheduling the report to publish it to the versioning system on the JReport Console page.

- When advanced running a report, in the Archive tab of the Advanced Run dialog:
  1. Specify where to save the result version by setting the Archive Location option.
     - To generate the report result version in the built-in folder, select **Built-in Version Folder**.
     - To generate report result version in a standalone resource node in the resource tree, select **My Reports Folder** or **Public Reports Folder**, and then provide the path and resource name information in the corresponding box.
  2. Apply an archive policy to the version as required and submit the task.

  Then, when the report finishes running, a report result version will be generated to the location specified.

- When scheduling a report to publish it to the versioning system, in the Publish > To Version tab of the Schedule dialog:
  1. Specify where to save the result version by setting the Archive Location option.
     - To generate the report result version in the built-in folder, select **Built-in Version Folder**.
     - To generate report result version in a standalone resource node in the resource tree, select **My Reports Folder** or **Public Reports Folder**, and then provide the path and resource name information in the corresponding box.
  2. Apply an archive policy to the version as required and submit the task.

  Then, when the schedule task is finished, a report result version will be generated to the location specified.

Creating dashboard versions

After a dashboard is saved into the server resource tree, you may want to update or modify it, and then save the changes, which will add a new version in the dashboard.

Notes:

- The resource path and name refers to the resource path and name in the resource tree. For instance, `/foldername/filename`.
- For The Public Reports folder option, the first slash mark (/) refers to the Public Reports folder in the resource tree, and the folder name (foldername) refers to an existing folder in the resource tree.
- For The My Reports folder option, the first slash mark (/) refers to the My Reports folder in the resource tree, and the folder name (foldername) refers to an existing folder in the resource tree.
- When generating report result versions in an existing standalone resource node in the resource tree, for example, creating a new version for a resource node, you should make sure to provide the path and name of the existing resource for The Public Reports folder or The My Reports folder option.

Browsing versions

To view the version information of a resource:

- For administrators, on the JReport Administration page, click **Resources** on the system toolbar and select **Resources** from the drop-down menu. In the Resources panel, browse to the resource, then click the **Version** button in the Control column.
- For end users, on the JReport Console > Resources page, browse to the resource, then do either of the following:
  - Select the resource row and click **Tools** > **Version** on the task bar of the Resources page.
  - Select the resource row, right-click in the row and select **Version** from the shortcut menu.
  - Put the mouse pointer over the resource row and click the **Version** command button on the floating toolbar.

The versions that a resource hosts are organized in the version table.

Version table

Relevant information about the versions that a resource hosts, such as the version date, version number, is collected and represented in a table, called the version table.

The version table is composed of the following columns:

- **Catalog Versions Table**

Column	Description
Version Date	The date and time of when the version was generated.
NLS Editor	Administrators can edit NLS for a specified catalog version by clicking the corresponding link.
Version Number	The serial ID that identifies a version in the version table.

- **Report Versions Table**

Column	Description
Version Date	The date and time of when the version was generated.
NLS Editor	Administrators can edit NLS for a specified report version by clicking the corresponding link.
Version Number	The serial ID that identifies a version in the version table.

- **Report Result Versions Table**

Column	Description
Result	The output file formats. You can click the links to view the output files.
Version Date	The date and time of when the version was generated.
Parameter File	The parameter file name. You can click the underlined file name to view the parameters.
Creator	The ID of the user who created the version.
Version Number	The serial ID that identifies a version in the version table.

- **Dashboard Versions Table**

Column	Description
Version Date	The date and time of when the version was generated.
Version Number	The serial ID that identifies a version in the version table.

- **Library Component Versions Table**

Column	Description
Version Date	The date and time of when the version was generated.
NLS Editor	Administrators can edit NLS for a specific library component version by clicking the corresponding link.
Version Number	The serial ID that identifies a version in the version table.

Tip: Some columns in the tables are not shown by default. To have them displayed, locate the table, click **Preferences** on the task bar if available, check the corresponding items in the Preferences dialog, then click **OK** to save the settings.

Applying an archive policy

JReport Server uses an archive policy to control resource versions. The archive policy can be applied to a single resource individually or to many resources in a folder as a whole. It can also be applied when you publish resources to the server resource tree or when you advanced run or schedule a report.

When applying an archive policy, you can choose whether to use multiple versions for a resource or always use the new version to replace the old one.

Archive as New Version

Specifies to use multiple versions for the specified resource.

- **Maximum Number of Versions**
  Specifies the maximum number of versions that will be listed in the version table of the resource. The default value is 0, which means that the version number is unlimited.

Replace Old Version

Specifies to replace the old version when a new version is generated.

If there is no archive policy specified for a resource, the resource will inherit the archive policy from its parent object. If afterwards you then specify an archive policy for the resource, the new policy will override the one inherited from the parent object.

Applying an archive policy to resources in the resource tree

To apply an archive policy to a resource in the resource tree, refer to the table below:

If you want to	Then do	Result
Apply archive policy when publishing resource	In the publish resource dialog (for how to access the dialog, see Publishing resources): • If the resource is to be published to the My Reports or Public Reports folder, set the Apply Archive Policy option as required. • If the resource is to be published to the My Components or Public Components folder, set the Maximum Number of Versions option as required.	The archive policy is applied to the resource. **Note:** A folder by itself does not have versions; the archive policy specified for a folder applies to the folder content.
Apply archive policy to a folder	1. Access the Properties dialog for the folder (for how to access the dialog, see Changing resource and folder properties). 2. In the dialog, ○ For folder in the My Reports or Public Reports folder (including the root folder itself), set the Apply Archive Policy option as required, then click **OK**. ○ For folder in the My Components or Public Components folder (including the root folder itself), set the Maximum Number of Versions option as required, then click **OK**.	The archive policy will be applied to all of the folder content. **Note:** This does not include resources that already have individually applied archive policies.
Apply archive policy to a resource	1. Access the Properties dialog for the resource (for how to access the dialog, see Changing resource and folder properties). 2. In the dialog, ○ For resource in the My Reports or Public Reports folder, set the Apply Archive Policy option as required, then click **OK**. ○ For resource in the My Components or Public Components folder, set the Maximum Number of Versions option as required, then click **OK**.	The archive policy is applied to the resource, overriding its inherited archive policy. **Note:** If you leave the archive policy unspecified, the resource will inherit the archive policy from its parent object, for example, the folder it resides in.
Apply archive policy when running a task in Advanced mode	1. On the JReport Console > Resources page, browse to the resource you want to run. 2. Put the mouse pointer over the resource row and click the **Advanced Run** button on the floating toolbar. 3. In the Archive tab, check the **Auto Archive Properties** option. 4. Finish the other relevant information, making sure that Archive Location is set to the resource tree folder. 5. Set the Apply Archive Policy option as required, and then click **Finish**.	The archive policy will be applied to a result type resource. **Note:** If you leave the Apply Archive Policy option unchecked, the resource will use its old archive policy or inherit the archive policy from its parent object, for example, the folder it resides in.
Apply archive policy when scheduling a task	1. On the JReport Console > Resources page, browse to the resource you want to schedule. 2. Put the mouse pointer over the resource row and click the **Schedule** button on the floating toolbar. 3. In the Publish tab, click the **To Version** sub tab, then check the **Publish to Versioning System** option. 4. Finish the other relevant information, making sure that Archive Location is set to the resource tree folder. 5. Set the Apply Archive Policy option as required, then click **Finish**.	The archive policy is applied to a result type resource. **Note:** If you leave the Apply Archive Policy option unchecked, the resource will use its old archive policy or inherit the archive policy from its parent object, for example, the folder it resides in.

Applying an archive policy to the built-in version table

The above table applies to the resource in the resource tree only. The versions in the built-in version folder are controlled by its own archive policy.

To apply an archive policy to the built-in version table, refer to the table below:

If you want to	Then do
Apply archive policy to a built-in version table	1. Access the version table for the resource (report type) (for how to access the table, see Browsing versions). 2. In the Report Result Versions tab, check the **Maximum Number of Versions** option, specify the versions to be saved as required, then click **OK**.
Apply archive policy when running a task in Advanced mode	1. On the JReport Console > Resources page, browse to the resource you want to run. 2. Put the mouse pointer over the resource row and click the **Advanced Run** button on the floating toolbar. 3. In the Archive tab, check the **Auto Archive Properties** option. 4. Finish the other relevant information, making sure that Archive Location is set to the Built-in Version Folder. 5. Set the Apply Archive Policy option as required, then click **Finish**.
Apply archive policy when scheduling a task	1. On the JReport Console > Resources page, browse to the resource you want to schedule. 2. Put the mouse pointer over the resource row and click the **Schedule** button on the floating toolbar. 3. In the Publish tab, click the **To Version** sub tab, then check the **Publish to Versioning System** option. 4. Finish the other relevant information, making sure that Archive Location is set to Built-in Version Folder. 5. Set the Apply Archive Policy option as required, then click **Finish**.

Deleting versions

After creating versions, periodically you may want to delete some expired or unused versions. You can choose to remove these versions manually or configure JReport Server to delete them automatically. When removing the versions using the user interface, the archive versions stored on disk are also physically deleted.

Deleting manually

To delete some versions of a resource manually, first open the version table of the resource (for how to access the table, see Browsing versions), then:

- For administrators, in the version table, check the checkbox ahead of the versions that you want to remove and then click the **Delete** link.
- For end users, in the version table, find the version you want to remove, then:
  - Select the version row and click **Edit** > **Delete** on the task bar.
  - Select the version row, right-click in the row and click **Delete** on the shortcut menu.
  - Put the mouse pointer over the version row and click the **Delete** button .

After receiving "The version has been deleted" message, view the version information again. You will find that the version you selected has now been removed from the version table.

Deleting automatically

There are two approaches to automatically deleting versions:

- **Apply Archive Policy**
  The Apply Archive Policy controls the number of versions that will be recorded in the version table of a resource.

  When creating a resource version, you can specify the maximum number that will be saved. If the number of versions exceeds the specified number, the oldest version will automatically be removed from the version list.

  For example, if you specify Maximum Number of Versions as 5, when the sixth version is created, the first version will automatically be removed.

- **Result Auto-delete**
  Result Auto-delete controls the duration of versions. It is only applicable to report result versions.

  When creating a report result version, you can specify a certain period of time to keep the version. The version will automatically be removed from the version list after the number of days or the specified date.

  For example, if you specify "Result Expires in 30 days", it will be automatically removed 30 days after its creation.

Advanced version topics

The following sections cover advanced version topics for administrators and developers. It will give you an understanding of version structure in depth.

- Version resource structure in the server database
- Storage of versions on disk
- Topics for developers

Version resource structure in the server database

Information about versions, folders, nodes, the security system, the completed table, and server runtime performance are all stored in the default database Derby. Also, since JReport Server provides multiple database support, you can configure your own database to store server data. For production systems it is recommended to use the same DBMS as your database application so the server data will be backed up with the rest of the application.

This section discusses how view resources were stored in the server database - Derby.

To open the Derby database, use Apache toolkit ij. ij is Derby's interactive JDBC scripting tool. It is a simple utility for running scripts or interactive queries against a Derby database. ij is a Java application that you start from a command window such as an MS-DOS Command Window or the Unix shell. ij provides several non-SQL commands for ease in accessing a variety of JDBC features for testing.

To open the defaultRealm.* files:

1. Start ij.bat in `<server_install_root>\derby\bin`.

2. Connect to database. To connect to a Derby database, you need to perform a valid database connection URL. ij automatically loads the appropriate driver based on the syntax of the URL. The following example shows how to connect defaultRealm by using the Connect command and the client driver:

   ```
   D:>java org.apache.derby.tools.ij
   ij version 10.5
   ij> connect 'jdbc:derby://localhost:8886/defaultRealm';
   ij>
   ```

3. After connected to the database successfully, ij allows the execution of Derby SQL statements interactively or via scripts.

   Note: All statements must be terminated with a semicolon.

   Example 1: Finding result version information from the database

   Execute the command `SELECT * FROM RESULTVERSION_2 WHERE CREATOR='admin';` to retrieve result versions which were created by the user 'admin'.

   Example 2: Finding catalog version information from the database

   Execute the command `SELECT * FROM CATALOGVERSION_3;` to fetch all catalog versions.

4. For more information about Derby and ij, please visit Apache website http://db.apache.org/derby/.

Note: Do not delete, update or insert data into the database. It is recommended that all modifications to the database be done from JReport Server instead of in the ij. Otherwise, it may result in JReport Server crashing if invalid data is found.

Storage of versions on disk

A working directory `<reporthome>\history` has been defined for use by the system database. Rather than store all the actual resources in the DBMS, the DBMS stores only pointers to the actual files. JReport Server uses this folder to store all of the parameter information and version files. By default, JReport Server creates 100 folders in its history folder (`<reporthome>\history`), and each of these folders can contain a further 3000 subfolders. These subfolders however cannot hold any further subfolders.

First, JReport Server puts the history information, such as archive versions and parameter files in folder 1. When the amount of subfolders in folder 1 reaches the maximum subfolder amount, it starts to put files in folder 2. When folder 2 is filled up, folder 3 will be used, then folder 4, folder 5, and so on, until all 100 folders have been filled up.

When all the 100 folders have been filled up with subfolders, JReport Server will then create another 100 folders, named 101 to 200, and will continue to store the history information in these folders, starting from folder 101. When the second 100 folders are full, another 100 folders will be created, and so on.

Changing the storage folder

JReport Server provides a server.properties file for you to configure the server. If you want to specify a folder other than `<reporthome>\history` to store the version files, you can set the folder for the resource.share.hist.dir property in this file.

The following describes the process of changing the storage folder:

1. After installing and starting JReport Server, the server.properties file is generated in the `<install_root>\bin` folder. You can then set the folder that your server data is to be stored in, in the resource.share.hist.dir property as `resource.share.hist.dir=D:\ServerData`.
2. Delete the properties, profiling and history sub folders from the `<install_root>` folder.
3. Restart JReport Server. The demo resources in the `<reporthome>\jreports` directory will now be stored in the specified ServerData folder.
4. After the above procedures have been performed, all parameter information and version files created will be stored in the specified ServerData folder.

Topics for developers

Other than implementation from the JReport Server UI, advanced users are also allowed to manage versions using the JReport Server API and RMI API.

The JReport Server API and RMI API is a set of Java programming interfaces that run reports, explore report resources, and provide access control for report servers. It is mainly used for writing servlets, JSPs and Java applications with report server features. You can find documentation that describes the usage of the JReport Server API in the section JReport Server APIs and Remote APIs.

Program examples to show how to publish a report to the versioning system, how to run a report, and how to publish a catalog/report can be found in the `<install_root>\help\samples\APIServer` folder.

Applying an archive policy

You may find that whenever you create catalog/report template versions, or report result versions, an archive policy is applied. The archive policy is a series of settings for controlling whether or not to use multiple versions for a specific resource, for specifying the maximum version amount and archive location, and for controlling whether or not to auto-delete versions after a certain period of time.

Archive policy parameters

Parameters	Values	Description	Parameters	Values	Description
APIConst. TAG_ENABLE_ARCHIVE_POLICY	true /false	Whether or not to apply a new archive policy.			
APIConst.TAG_AUTO_ARCHIVE	true /false	Whether or not to auto-archive the viewed result.			
APIConst. TAG_ARCHIVE_LOCATION	0	Built-in version folder is used as the result archiving location.			
	1	The My Reports folder is used as the result archiving location.			
			APICONST. TAG_ARCHIVE_MY_DESTINATION		Sub folder name in the My Reports folder for this user. For example, /rtp100, which means to output the file to /USERFOLDERPATH/<userid>/rtp100.
	2	The Public Reports folder is used as the result archiving location.			
			APICONST. TAG_ARCHIVE_PUBLIC_DESTINATION		Sub folder name in the Public Reports folder for all the users. For example, /ActimizeTest/rtp100
APIConst. TAG_REPLACE_OLD_VERSION	true /false	Whether or not to replace the previous version.			
APIConst. TAG_ARCHIVE_NEW_VERSION	true /false	Whether or not to archive the next version as a new version.			
APICONST. TAG_NEED_MAXVERSION	0 or N	The number of versions to keep for this resource result.			
APICONST.TAG_NEED_EXPIRE	true /false	Whether or not to auto delete this version.			
APIConst.TAG_EXPIRE_METHOD	0	Version expires after a number of days.			
			APIConst.DAY_EXPIRE	Number	The number of days until a version expires.
	1	Version expires after a certain date.			
			APIConst.TAG_EXPIRE_YEAR	Number	The year of expiration.
			APIConst.TAG_EXPIRE_MONTH	Number	The month of expiration.
			APIConst.TAG_EXPIRE_DATE	Number	The date of expiration.

Scheduling report result to the versioning system

You can use the Java class HttpRptServer with the following methods to schedule report result to the versioning system: runTask(), and submitScheduledTask().

There is a Hashtable argument for these two methods, which includes the parameters used for running tasks. Here are some parameters that you will need when specifying your Hashtable for scheduling to version:

Parameters	Values	Description	Parameters	Values	Description
APICONST. TAG_TO_VERSION	true /false	Whether or not to schedule to the versioning system.			
			APICONST. TAG_TO_VERSION_EXCEL	true /false	Whether or not to schedule to an Excel version.
			APICONST. TAG_TO_VERSION_HTML	true /false	Whether or not to schedule to an HTML version.
			APICONST. TAG_TO_VERSION_PDF	true /false	Whether or not to schedule to a PDF version.
			APICONST. TAG_TO_VERSION_PS	true /false	Whether or not to schedule to a PostScript version.
			APICONST. TAG_TO_VERSION_RSD	true /false	Whether or not to schedule to a Page Report Result version.
			APICONST. TAG_TO_VERSION_RST	true /false	Whether or not to schedule to an rst version.
			APICONST. TAG_TO_VERSION_RTF	true /false	Whether or not to schedule to an rtf version.
			APICONST. TAG_TO_VERSION_TXT	true /false	Whether or not to schedule to a TEXT version.
			APICONST. TAG_TO_VERSION_XML	true /false	Whether or not to schedule to an XML version.

Obtaining version information

You will be able to fetch version records with the following function calls:

1. Get the resource manager from the HttpRptServer:

   public ResourceManager getResourceManager()

2. Then use the following methods to obtain a version:
   - **Report version**
     getReportVersions(String userID, String rptName)
   - **Report Result version**
     getResultVersion(String userID, String rptName, int versionNumber)
   - **Catalog version**
     getCatalogVersions(String userID, String catName)
   - **Result version**
     getResultDocVersions(String userID, String resultDocName)
   - **Dashboard version**
     getReportVersions(String userID, String dashboardName)
   - **Library component version**
     getLCVersions(String userID, String lcName)

Note: userID is only used for checking privilege in this method, and not for filtering the submitter.

Managing security

JReport Server provides a security system which allows you to set up and maintain security on the server and protect your resources from inappropriate access by users.

This section focuses on accomplishing security management in JReport Server by means of setting up roles, users, and assigning certain permissions.

You can find other information related to server security in the sections Security and Server security system.

This section covers the following topics:

- Managing realms
- Managing user accounts
- Managing groups
- Managing roles
- Managing privileges
- Managing aliases
- Managing dynamic security
- Managing organizations

Managing realms

To manage realms, you must be a member of the administrator role in order to access the JReport Administration page. Then, on the JReport Administration page, click **Security** on the system toolbar and select **Realm** from the drop-down menu to go to the Realm panel, where you can manage the realms as required.

The following topics detail how to manage a realm:

Creating a new realm

1. Click the **Create a New Realm** link.
2. In the New Realm dialog, type a name for the realm, and select an authentication mode as the scheme.

   Basic Authentication uses the base64 encryption method as defined by RFC 1945. Digest Authentication uses the MD5 encryption method as defined by RFC 2069. Basic Authentication is lower security but is usually what is used on intranets. Digest Authentication provides much higher security and is usually used with SSL in a highly secure environment.
3. When done, click **OK**, and the realm will then be added to the realm list table.

When a new realm is created, it is assigned with built-in users, groups, and the default resource tree. Remember to activate the correct realm before allowing clients to visit.

Activating a realm

The realm must be activated before its content, such as resources, users, groups, and roles, and it can then be accessed by the client users. There can be only one active realm at any time.

To activate a realm:

1. On the JReport Administration page, click **Configuration** on the system toolbar, and then click **Service** from the drop-down menu.
2. Select the realm you want to activate from the Active Realm drop-down list.
3. Click **Save** to apply the settings.
4. Restart JReport Server so that the changes can take effect.

Managing the users, groups and roles in an inactive realm

Users, groups and roles are only available when the realm they belong to is active. However, users, groups and roles in an inactive realm can still be managed by users in the administrator role.

To manage users, groups and roles in an inactive realm:

1. Click the **Select** link in the Control column of the realm list table to select the realm you want to manage.

   All realms in JReport Server are listed in the realm list table. The State column shows the status of the realms. An active realm is marked as Active Realm. A selected realm is marked as Selected Realm. If the realm is both active and selected, it is marked as Active Realm.
2. Click **Security** and then **User**, **Group** or **Role** to manage the selected realm.

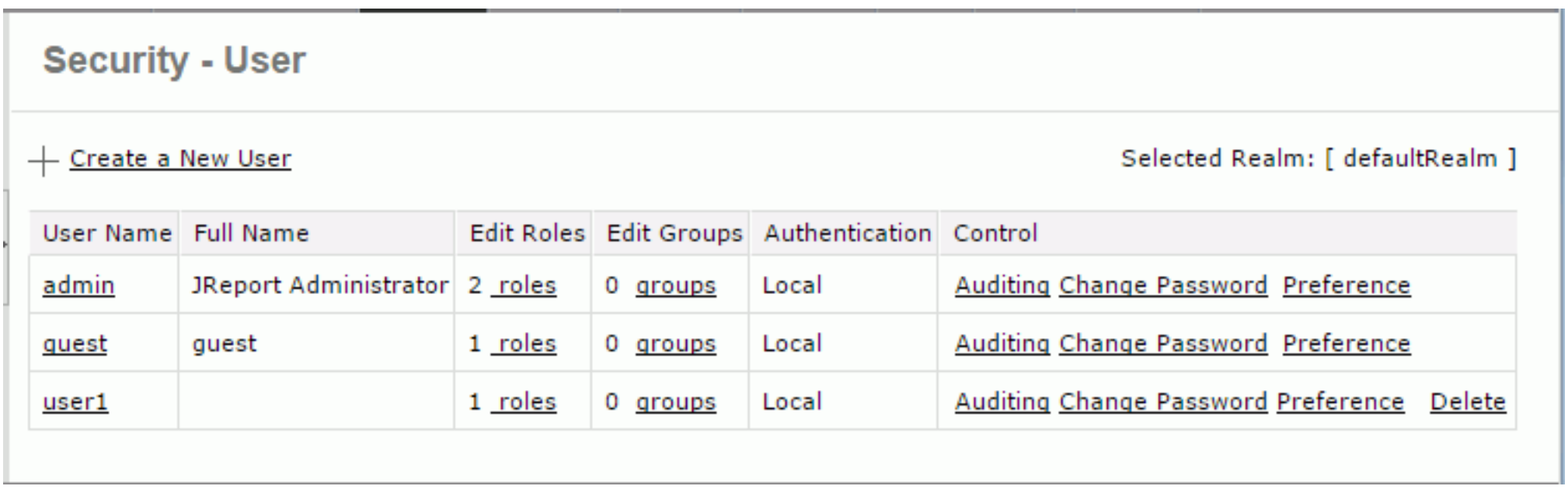
Information about users, groups and roles in the selected realm is listed in the User, Group, and Role panels.

Deleting a realm

If you find a realm is no longer required, you can delete it by clicking the corresponding Delete link in the Control column of the realm list table.

Managing user accounts

To manage user accounts, you must be a member of the administrator role in order to access the JReport Administration page. Then, on the JReport Administration page, click **Security** on the system toolbar and select **User** from the drop-down menu to display the Use panel, where you can manage the user accounts as required.



The following topics describe how to manage a user account.

Creating a new user account

1. In the Security > Realm panel, select a realm to which you want to add the user.
2. In the Security > User panel, click the **Create a New User** link.
3. In the New User dialog, provide the information for the user as required.
4. Click **OK**, and the user will then be added to the user account table.

Modifying a user account

1. In the Security > Realm panel, select the realm in which the user is.
2. In the Security > User panel, click the name of the user.
3. In the Edit User dialog, edit the user information as required.
4. When done, click **OK** to accept the changes.

Adding roles to a user

A user can be assigned more than one role. A user that holds multiple roles has all the privileges that these roles have.

To add roles to a user:

1. In the Security > Realm panel, select the realm in which the user is.
2. In the Security > User panel, click the **role(s)** link of the user.
3. In the displayed page, click the **Add Roles** link.
4. In the role table, check the roles that you want to add to the user, then click the **Add** button.
5. Click **Security** > **User** to return to the user list.

Adding a user to groups

A user can be assigned to more than one group. A user that belongs to multiple groups has all the privileges that these groups have.

To add a user to some groups:

1. In the Security > Realm panel, select the realm in which the user is.
2. In the Security > User panel, click the **group(s)** link of the user.
3. In the displayed page, click the **Add Groups** link.
4. In the group table, check the groups that you want to add the user to, then click the **Add** button.
5. Click **Security** > **User** to return to the user list.

Removing roles/groups from a user

If you want to remove certain roles a user holds, or some groups a user belongs to from a user, follow the steps below:

1. In the Security > Realm panel, select the realm in which the user is.
2. In the Security > User panel, click the **role(s)**/**group(s)** link of the user.
3. In the roles/groups table of the user, check the roles/groups you want to remove, then click the **Remove** button.

Once a role or group is removed from a user, the user will no longer have the privileges the role or group has.

Auditing a specific user

You can have the server to audit a user, and the resulting information will be written into the log files.

To audit a user:

1. In the Security > Realm panel, select the realm in which the user is.
2. In the Security > User panel, click the **Auditing** link of the user.
3. In the Auditing dialog, specify the events which you want to have audited for this user.
4. When done, click **OK** to accept the changes.

Changing the password of a user

1. In the Security > Realm panel, select the realm in which the user is.
2. In the Security > User panel, click the **Change Password** link of the user.
3. In the Change Password dialog, specify the password of the current logged in user.
4. Specify the new password for the user and confirm it by entering it a second time.
5. When done, click **OK** to accept the changes.

Setting user preferences

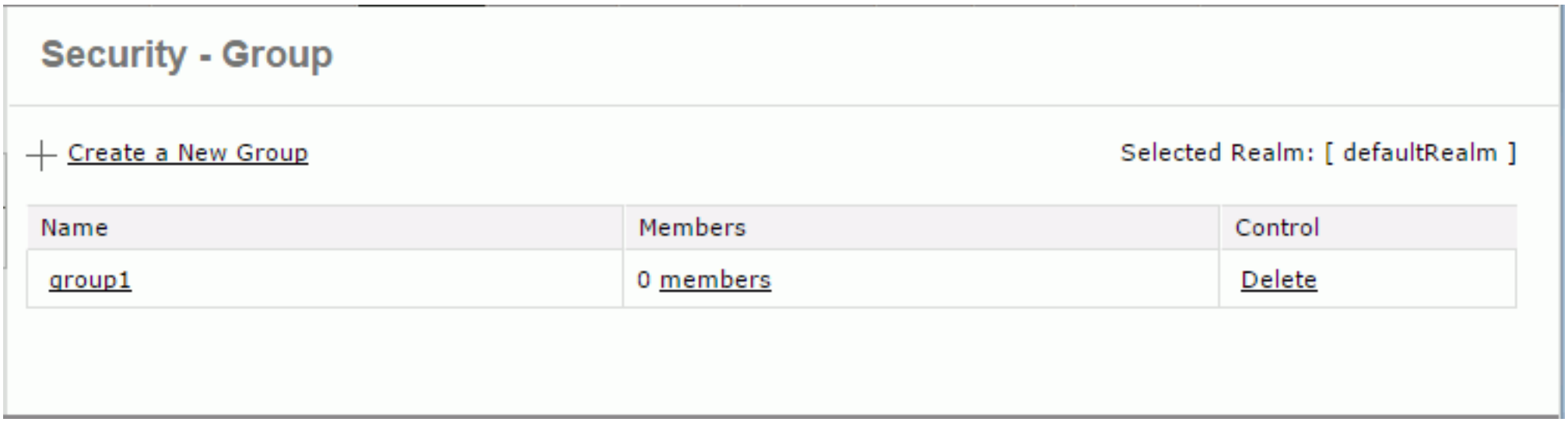
1. In the Security > Realm panel, select the realm in which the user is.
2. In the Security > User panel, click the **Preferences** link of the user.
3. In the Preferences dialog, specifies the server preferences and Page Report Studio preferences for the user accordingly.
4. When done, click **OK** to accept the changes.

Deleting a user account

If you find a user account is no longer required, you can delete it by clicking the corresponding Delete link in the Control column of the user account table. However, the built-in user accounts, such as admin and guest, and users that hold roles other than the everyone role, or that belong to any group, cannot be deleted. A user cannot delete himself from the user list either.

Managing groups

To manage groups, you must be a member of the administrator role in order to access the JReport Administration page. Then, on the JReport Administration page, click **Security** on the system toolbar and select **Group** from the drop-down menu to display the Group panel, where you can manage the groups as required.



The following topics explain how to manage a group.

Creating a new group

1. In the Security > Realm panel, select a realm to which you want to add the group.
2. In the Security > Group panel, click the **Create a New Group** link.
3. In the New Group dialog, specify the settings for the new group.
4. When done, click **OK**, and the new group will then be added into the group list table.
5. Click **Security** > **Group** to return to the group list table.

Modifying a group

1. In the Security > Realm panel, select the realm in which the group is.
2. In the Security > Group panel, click the name of the group.
3. In the Edit Group dialog, edit the group information as required.
4. When done, click **OK** to accept the changes.

Editing members of a group

You can edit members of a group, such as adding a new member, or removing a member from the group.

To edit the members in a group:

1. In the Security > Realm panel, select the realm in which the group is.
2. In the Security > Group panel, browse to the specific group, and then click the **member(s)** link.
3. Edit the members of the group as follows:
   - To remove a member from the group, check the member and then click the **Remove** button.
   - To add a member to the group, click the **Add Members** link, check the new member, and then click the **Add** button.

Notes:

- A group can have more than one child member and parent member.
- A parent member cannot be added to the current group as its child member.

Deleting a group

If you find a group is no longer required, you can delete it by clicking the corresponding Delete link in the Control column of the group list table. However, groups that are not empty, having child members or parent members, cannot be deleted.

Managing roles

To manage roles, you must be a member of the administrator role in order to access the JReport Administration page. Then, on the JReport Administration page, click **Security** on the system toolbar and select **Role** from the drop-down menu to display the Role panel, where you can manage the roles as required.

The following topics describe how to manage a role.

Creating a new role

1. In the Security > Realm panel, select a realm to which you want to create the role.
2. In the Security > Role panel, click the **Create a New Role** link.
3. In the New Role dialog, specify the settings for the role.
4. Click **OK**, and the new role will then be added to the role list table.

Modifying a role

1. In the Security > Realm panel, select the realm in which the role is.
2. In the Security > Role panel, click the name of the role.
3. In the Edit Role dialog, edit the role information as required.
4. When done, click **OK** to accept the changes.

Editing members of a role

You can edit the members of a role, such as adding a new user or role, or removing a member from the role.

To edit the members in a role:

1. In the Security > Realm panel, select the realm in which the role is.
2. In the Security > Role panel, browse to the specific role, and then click the **member(s)** link.
3. Edit the members of the role as follows:
   - To remove a member from the role, check the member and then click the **Remove** button.
   - To add a member to the role, click the **Add Members** link, check the new member, and then click the **Add** button.

Notes:

- A role can have more than one child group, child role and parent role.
- A parent role cannot be added to the current role as its child.
- Some members cannot be removed from the role they belong to, such as admin in the administrator role and guest in the everyone role. A user cannot remove himself from the administrators role.

Deleting a role

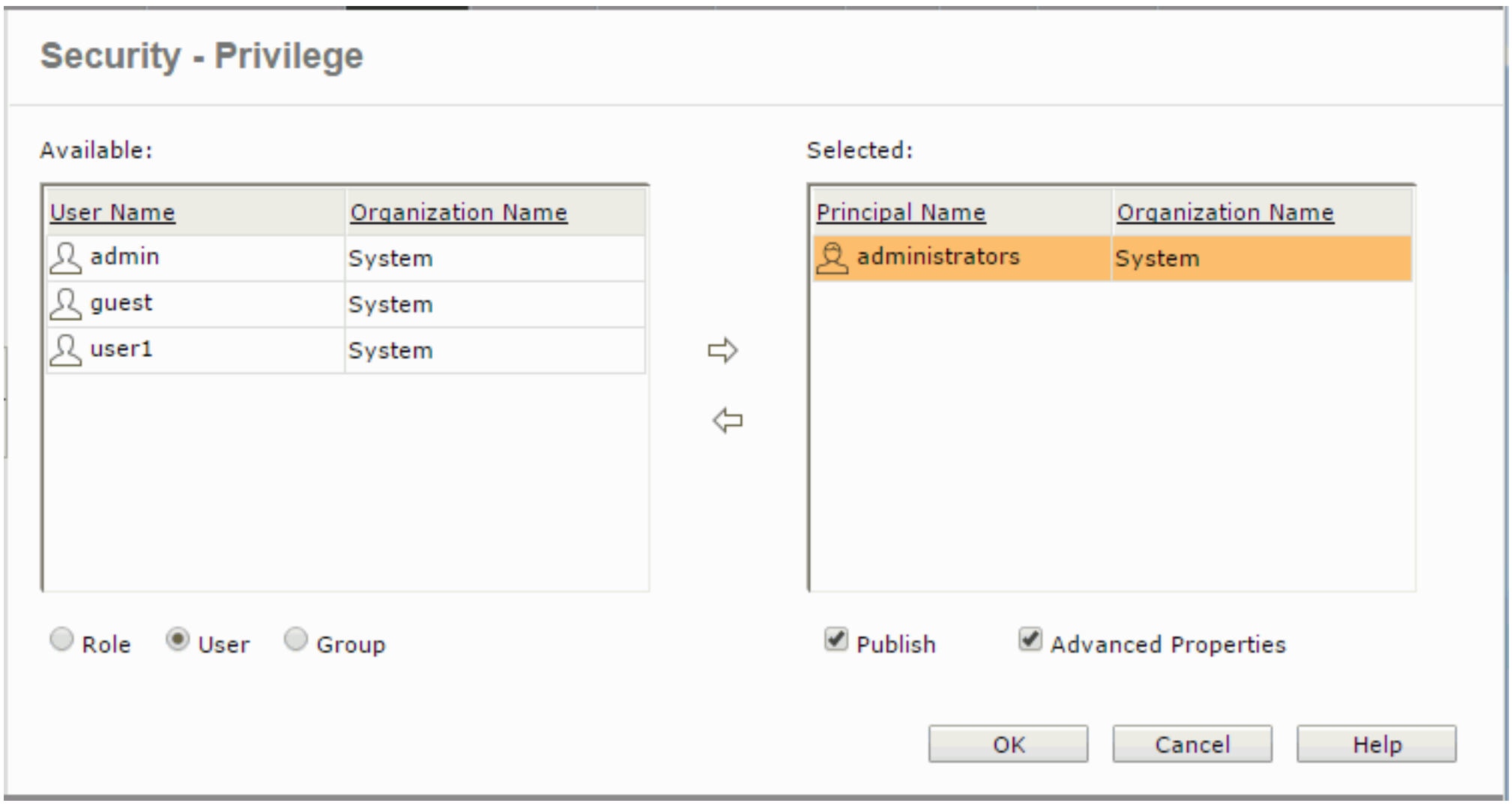
If you find a role is no longer required, you can delete it by clicking the corresponding Delete link in the Control column of the role list table. However, the built-in roles, such as administrators and everyone, and roles that are not empty, cannot be deleted.

Managing privileges

JReport Server offers these types of privileges for users, groups, and roles: Publish and Advanced Properties. Users that are granted the Publish privilege will be able to publish resources to JReport Server, while users that have the privilege of Advanced Properties are allowed to view advanced information of version properties such as catalog connections and report related resources.

To manage privileges, you must be a member of the administrator role in order to access the JReport Administration page. Then,

1. On the JReport Administration page, click **Security** on the system toolbar and select **Realm** from the drop-down menu.
2. In the Realm panel, select the realm which contains the roles/groups/users you want to grant privilege for.
3. Click **Security** on the system toolbar and select **Privilege** from the drop-down menu to display the Privilege panel.



You can now grant and remove role/group/user privileges in the panel.

Granting role/group/user privileges in the Privilege panel

- **For roles/groups/users that already have privileges**
  Roles/groups/users that already have privileges are listed in the Selected table. To assign privileges to them, select a role/group/user in the table, then check the checkbox beside the type of privilege you want to grant. Then click **OK** to apply the changes.
- **For roles/groups/users that have neither of the privileges**
  1. In the Privilege panel, to assign privileges to a role/group/user, select the **Role**/**Group**/**User** radio button accordingly.
  2. Select the role/group/user in the Available table and click to add it to the Selected table.
  3. The newly added role/group/user is selected by default. Check the type of privilege you want to grant.
  4. Click the **OK** button to apply the changes.

Removing privileges from a role/group/user in the Privilege panel

To remove privileges from a role/group/user, you can choose either of the following two methods:

- To remove a privilege from a role/group/user, select the role/group/user in the Selected table, uncheck the checkbox beside the privilege, then click **OK** to apply the changes.
- To remove all privileges from a role/group/user, first select the role/group/user in the Selected table, then click . By doing this, the role/group/user will be deleted from the Selected table.

Granting and removing role/group/user privileges in the Role/Group/User panel

1. In the Security > Realm panel, select the realm in which the role/group/user is.
2. In the Security > Role/Group/User panel, click the underlined role/group/user name.
3. In the Edit Role/Edit Group/Edit User dialog, check or uncheck the target privileges.
4. Click the **OK** button to apply the changes.

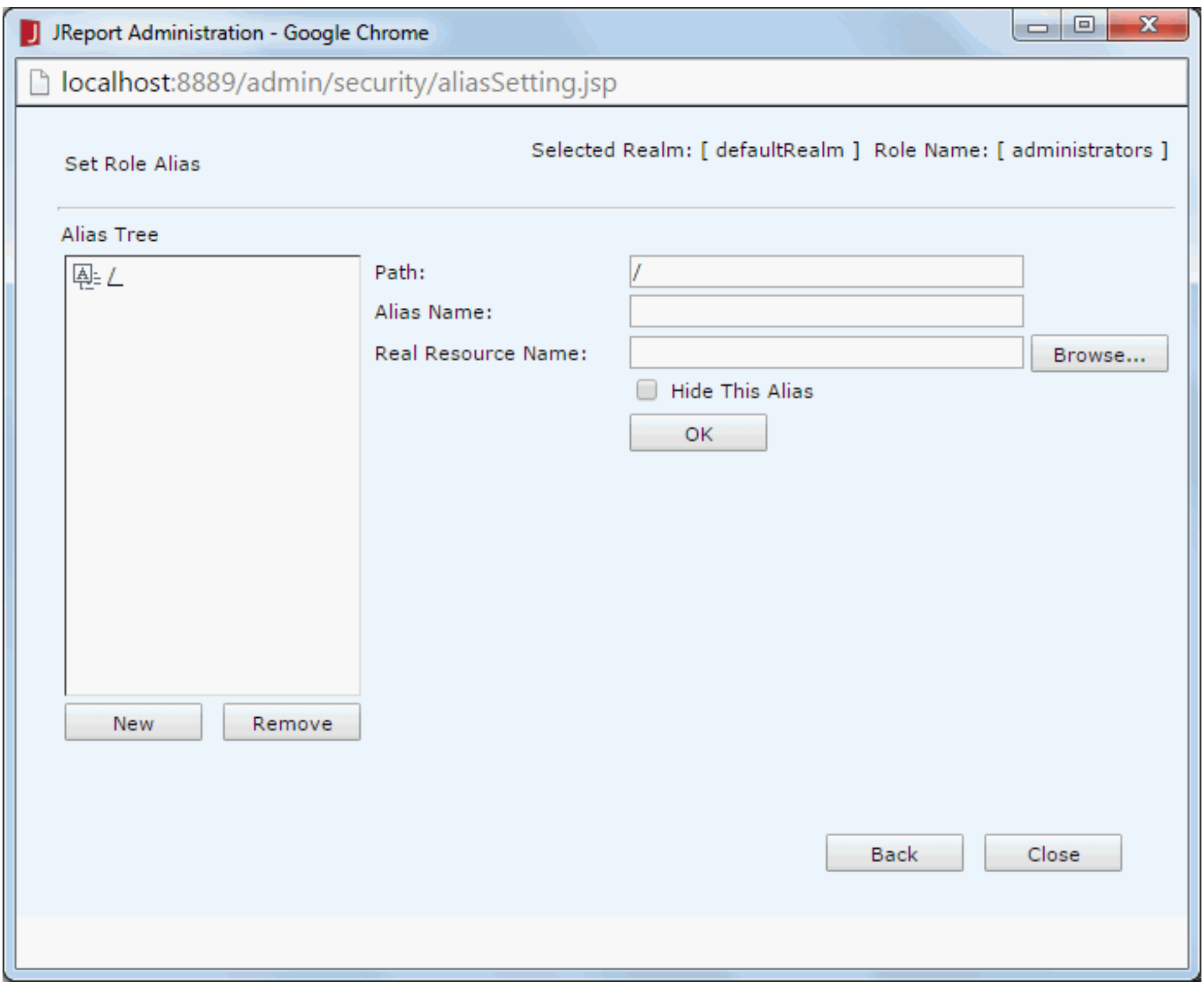
Managing aliases

JReport Server organizes its files and directories into a Resource Tree. Aliases are used to provide certain "views" of the tree, and allow different users to have different views. For example, you can set an alias resource tree (based on the resource tree) for Mary in Sales, which enables her to only see the Sales resource node, allowing her direct access to the report files she is interested in. An alias is a combination of users and the resource node.

However, in order to manage aliases, you must be a member of the administrator role in order to access the JReport Administration page.

Setting an alias resource tree for a role, group, or user

1. Log onto the JReport Administration page.
2. Click **Security** on the system toolbar and select **Realm** from the drop-down menu.
3. Select the realm in which the role, group, or user is.
4. Click **Security** on the system toolbar and select **Alias** from the drop-down menu.
5. Set an alias resource tree for a role/group/user, click the corresponding set alias link.
6. Select a role/group/user, and then click the **Next** button to open the Set Alias dialog.



7. Click the **New** button to create a new alias node in the root node.
8. In the Alias Name field, replace the default name newAlias with a name for the alias.
9. Click the **Browse** button to specify a destination resource from the server resource tree that is to be associated with the new alias node. If you do not want the alias node to be shown in the alias resource tree, check the **Hide This Alias** option.
10. Click the **OK** button to complete the creation of the new alias node.
11. Use the following steps to create further alias nodes:
    a. Select the node in which you want to create a new alias node.
    b. Click **New** to create.
    c. In the Alias Name field, replace the default name newAlias with a name for the alias.
    d. Click the **Browse** button to specify a destination resource from the server resource tree that is to be associated with the new alias node.
    e. If you do not want the alias node to be shown in the alias resource tree, check **Hide This Alias**.
    f. Click the **OK** button to complete the creation of the new alias node.
12. After you have finished with the alias resource tree, click the **Close** button to exit the dialog.

    You will now see that the just set role/group/user is listed in the corresponding alias list in the Alias panel.

Viewing the alias resource tree of a role, group, or user

1. In the Security > Realm panel, select the realm in which the role, group, or user is.
2. In the alias list of the Security > Alias panel, select the underlined name of the role/group/user you want to view.
3. The Set Alias dialog will then be opened for you to view the alias resource tree information of the role/group/user.

Editing the alias resource tree of a role, group, or user

1. In the Security > Realm panel, select the realm in which the role, group, or user is.
2. In the alias list of the Security > Alias panel, select the underlined name of the role/group/user you want to edit.
3. The Set Alias dialog will then be displayed for you to edit the alias resource tree of the role/group/user. You can edit by hiding/unhiding an alias node, creating a new alias node, or removing an existing alias node.
   - **To hide / unhide an alias node:**
     a. Select the alias node in the alias resource tree.
     b. Check/uncheck the Hide This Alias option, and then click **OK**.
   - **To create a new alias node:**
     a. Select the alias node in which you want to create a new alias node.
     b. Click **New**.
     c. In the Alias Name field, replace the default name with a name.
     d. Click the **Browse** button to search for the destination resource from the Server resource tree that is to be associated with the alias node.
     e. Click **OK** to create the alias node.
   - **To remove an existing alias node:**
     a. Select the alias node from the alias resource tree.
     b. Click **Remove** to remove the alias node.

Notes:

- An alias tree is based on the resource nodes (not virtual resource nodes) of the resource tree.
- By default, the alias resource tree root for each user refers to the resource tree root.
- When an alias tree is activated for a user, all resource access is then controlled by the alias resource tree.

Managing dynamic security

Administrators can create dynamic security policies for catalogs published to the server using security files, which contain catalog internal security definitions. With dynamic security, administrators can dynamically change the security policy applied to a catalog.

Creating security files

The security files that can be used to create dynamic security on JReport Server are in XML format. You can create the security files by yourself, however, the best way to generate a security file is using JReport Designer.

A security file can contain security definitions of record-level security (RLS), column-level security (CLS) and member-level security (MLS). For how to define RLS and CLS in an XML file, see the topic Importing/exporting security information from/to external XML files in the *JReport Designer User's Guide*. As for the format of MLS, you can refer to the sample file DynamicSecurity.xml, which is stored in `<install_root>\help\samples\DynamicSecurity`.

Suppose that an MLS policy is defined as follows:

Principal	Element	Visible Permission	Access Permission	Allowed Set	Denied Set	Allow Unspecified Members
everyone	Customer.Country	Allow	Allow	China	Japan	True

In the XML file, it will be written like below:

```
- <BVACL>
    <Role>everyone</Role>
    <BVGroupDisplayName>Customers.Country</BVGroupDisplayName>
    <ElementVisiblePermission>Allow</ElementVisiblePermission>
    <ElementAccessPermission>Allow</ElementAccessPermission>
    <MemeberPermission AllowUnspecifiedMembers="True" />
  - <AllowValueList>
    - <BinaryOprandExp>
        <Operator>=</Operator>
        <ConstantExp SqlType="String" Value="China" />
      </BinaryOprandExp>
    </AllowValueList>
  - <DenyValueList>
    - <BinaryOprandExp>
        <Operator>=</Operator>
        <ConstantExp SqlType="String" Value="Japan" />
      </BinaryOprandExp>
    </DenyValueList>
  </BVACL>
```

When the Allowed Set and Denied Set are defined using expressions, for example:

Principal	Element	Visible Permission	Access Permission	Allowed Set	Denied Set	Allow Unspecified Members
everyone	Orders Detail.Order ID	Allow	Allow	"Order ID" > 3431 And "Order ID" < 3005	"Order ID" between 3008 and 3011	True

In the XML file, the MLS policy will be like this:

```
- <BVACL>
    <Role>everyone</Role>
    <BVGroupDisplayName>Orders Detail.Order ID</BVGroupDisplayName>
    <ElementVisiblePermission>Allow</ElementVisiblePermission>
    <ElementAccessPermission>Allow</ElementAccessPermission>
    <MemeberPermission AllowUnspecifiedMembers="True" />
  - <AllowExp>
    - <LogicExp>
        <Operator>And</Operator>
      - <BinaryOprandExp>
          <Operator>></Operator>
          <ConstantExp Value="3431" />
        </BinaryOprandExp>
      - <BinaryOprandExp>
          <Operator><</Operator>
          <ConstantExp Value="3005" />
        </BinaryOprandExp>
      </LogicExp>
    </AllowExp>
  - <DenyExp>
    - <TripleOprandExp>
        <Operator>between</Operator>
        <ConstantExp Value="3008" />
        <ConstantExp Value="3011" />
      </TripleOprandExp>
    </DenyExp>
  </BVACL>
```

Creating dynamic security

To create a dynamic security policy, take the following steps:

1. Log onto the JReport Administration page.
2. Click **Security** on the system toolbar and select **Data Source** from the drop-down menu to display the Dynamic Security panel.
3. In the Dynamic Security panel, click **New Dynamic Security** on the task bar.
4. In the New Dynamic Security dialog, click the **Browse** button next to the Catalog text field.
5. In the Select Catalog dialog, browse to the catalog you want to create dynamic security for and click **OK**.
6. Click the **Browse** button next to the Security File text field, select the security file based on which the dynamic security for the catalog will be created, then click **Open**.
7. Click **OK** in the dialog to create the dynamic security, and it is added to the dynamic security table.

One catalog can have several dynamic security policies. In this case, you can specify which one of them will be used as the default dynamic security of the catalog. To do this, click the **false** link in the Is Default column for the dynamic security.

If you want to remove a dynamic security policy, check it in the dynamic security table, then click the **Delete** link above the table. You can delete multiple dynamic security policies at a time.

Changing the security file of a dynamic security policy

The security file based on which the dynamic security policy of a catalog is created can be modified. To do this, click the **Upload** link in the Controls column for the dynamic security, then in the Upload Dynamic Security dialog, select another security file to create the dynamic security for the catalog.

Downloading dynamic security

The dynamic security defined on server can be downloaded to a file. To do this, click the **Download** link in the Controls column for the dynamic security, in the displayed dialog, click **OK** to save it. You can then further modify the security definitions in the file. After editing, you can upload the file to a dynamic security policy to change the security definitions in the policy.

Applying dynamic security

When a catalog is specified with a default dynamic security policy, when running reports, dashboards, or Visual Analysis templates which use the catalog, no matter interactive, on-demand, or scheduled, JReport will always try to load the dynamic security for the catalog before running and try to use the security defined in the security file on which the dynamic security is created to replace the catalog security.

- **For business view**
  For each given principal and business view, if there is security definition in the security file, the whole security definition for this principal on this business view will be replaced.

  If the security file mentions a business view element, but the element is not in the catalog, it will be ignored.
- **For security entry**
  For a given principal and a security entry, if there is CLS definition in the security file, it will replace the CLS in the catalog security. If there is RLS definition in the file, it will replace the RLS in the catalog security.

If there is no default dynamic security specified for a catalog, the security definitions created in the catalog at design time will be used.

Managing organizations

JReport Server supports the multi-tenancy architecture by organizing users into different organizations. An organization is a group of users that has its own administrator.

Organization is a separately licensed feature of JReport Server. It is installed together with JReport Server so only the license key needs to be updated to enable it to work. To find out how to license Organization, please contact Jinfonet sales at sales@jinfonet.com or contact your Enterprise Account Manager.

When organization is enabled, you need to specify the organization name before logging in JReport Server. The available organizations are listed in the drop-down list for selection. The organization item *System* means that the login user is a non-organization user. For organization users, the correct organization name must be provided, otherwise they cannot log in. When organization is disabled, only non-organization users can log in.

Organization definitions

There are two types of admin in the server security system:

- **System admin**
  The administrator who has the administrators role. System admin must be a non-organization user.
- **Organization admin**
  The administrator who has the administrators role in an organization. You cannot log onto JReport Server Monitor as an organization admin. Only the system admin can log onto JReport Server Monitor and see both organization and non-organization information.

There are two types of users in the server security system:

- **Non-organization users**
  Belong to no organizations. When upgrading from a version earlier than version 13, the users, groups, and roles in the earlier version are all regarded as non-organization principals.
- **Organization users**
  Users, groups, and roles created in an organization are called organization users/groups/roles.

  An organization user/group/role can belong to any other non-organization group/role, but a non-organization user/group/role cannot belong to any organization group/role. An organization user/group/role cannot belong to any group/role of other organizations.

  Organization admin cannot add users/groups/roles to non-organization groups/roles.

  The naming rule of an organization user is composed of the organization name and the user name separated by "\". The format is [Organization Name]\[User Name]. For example, a\b, here "a" is the organization name and "b" is the user name.

  If organization is disabled, the organization users/groups/roles will not be supported. However the information will be saved in the server and can be retrieved next time when organization is enabled again.

There are two types of public resources in the server resource system:

- **System public resources**
  Public resources that can be accessed by all organization users and non-organization users. They are located in the Public Reports and Public Components folders.
- **Organization public resources**
  Each organization has its own public resources which can only be accessed by the users within the organization. Organization admin has built-in permissions on the organization public resources the same as system admin on the system public resources. Organization public resources are located in the Organization Reports and Organization Components folders.

Within each organization, there is a built-in admin and two built-in roles:

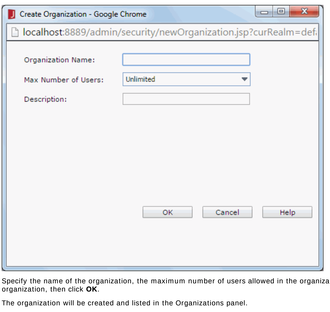
- **Built-in admin**
  - The full name of the default admin is organization administrator, also as organization admin.
  - The organization admin's user name is *admin* and password is *admin*. The password could be changed by the organization admin himself. If the password is forgotten, the system admin can retrieve it since the system admin can reset password for all users.
  - The organization admin owns the administrators role and everyone role by default, and it must always have the administrators role.
- **Two built-in roles**
  - **administrators**
    It has no parent role by default. It enables Publish and Advanced Properties privileges by default.
  - **everyone**
    It has no parent role by default. It disables Publish and Advanced Properties privileges by default.

Creating organizations

The system admin can create and delete organizations.

To create an organization:

1. On the JReport Administration page, click **Configuration** on the system toolbar and then select **Organizations** from the drop-down menu.
2. In the Organizations panel, click the **Create** link. The Create Organization dialog appears.



3. Specify the name of the organization, the maximum number of users allowed in the organization, and the description about the organization, then click **OK**.

   The organization will be created and listed in the Organizations panel.

In the Organizations panel, the system admin can further edit the maximum number of users in the organizations, or delete the organizations that are not required.

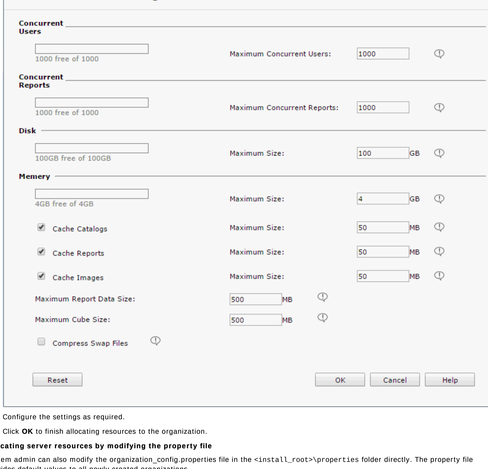
- To edit the maximum number of users in an organization, double-click its number in the Max Number of Users column to enter the edit mode. Select a value from the drop-down list or input an integer number in the combo box directly, then click outside of the combo box to accept the change. The **x** in the combo box is used to clear the input text.
- To delete an organization, select the checkbox ahead of the organization and then click **Delete**. To delete all the organizations at a time, select the checkbox on the column header and then click **Delete**.

Allocating server resources to organizations

System admin can allocate server resources to different organizations, then when one organization encounters report running performance problems, the other organizations will not be affected. The server resources include maximum concurrent users/reports, maximum disk/memory size, whether to cache catalogs, reports, or images used in the organization and the maximum memory size for them as well as for cached report data and cube, and whether to compress swap files so as to reduce I/O time by increasing CPU usage.

Allocating server resources via UI

1. In the Organizations panel, click the **Resource Allocation** link in the Control column of the organization in the organization table. The Resource Allocation dialog appears.



2. Configure the settings as required.
3. Click **OK** to finish allocating resources to the organization.

Allocating server resources by modifying the property file

System admin can also modify the organization_config.properties file in the `<install_root>\properties` folder directly. The property file provides default values to all newly created organizations.

For information about which property is mapped to which option in the Resource Allocation dialog, see Appendix 4: Mapping list of UI options and properties.

JReport Administration page for organization admin

After the system admin create an organization, the organization admin of the organization will be able to log in JReport Server by the default user name *admin* and the default password *admin* to manage resources and users in this organization and configure JReport Server to adapt for the organization.

The JReport Administration page contains the following tabs on the system toolbar for organization admin:

- **Resources**
  The Resources tab contains two additional folders under the root directory: *Organization Reports* and *Organization Components*. They are used to put reports and library components used by the users in the organization.

  Organization admin can publish resources to the My Reports/Components or Organization Reports/Components folder. For the Public Reports/Components folder, organization admin cannot publish resources, update properties, delete resources, or edit NLS. Meanwhile, the feature of using resources from real path is disabled.
- **Configuration**
  The Configuration tab contains the following items:
  - Performance
    Organization admin can define which catalogs and reports to be preloaded based on the resource allocation.
  - Dynamic Connections
    Organization admin can define dynamic connections for users in the organization.
  - Dynamic Display Names
    Organization admin can define dynamic display names for users in the organization.
- Security
  The Security tab contains these items: User, Group, Role, Privilege, and Dynamic Security. Organization admin can only see the users/groups/roles in the organization and cannot change the realm.
- Cube
  In the Cube tab there is no Configuration link since the value is retrieved from resource allocation. Organization admin can define cube based on the catalogs used within the organization only.
- **Cache**
  The Cache tab contains the following items:
  - Data Cache
    There is no Cache Configuration link. Organization admin can cache data based on the catalogs used within the organization only.
  - Report Cache
    Organization admin can define which catalogs and reports and how many of them to be cached.

JReport Console page for organization users

The Resources tab on the Console page contains two additional folders under the root directory: *Organization Reports* and *Organization Components*. They are used to put reports and library components used by users in the organization.

Organization users cannot see any information about physical disk or import resources from real path. For the Public Reports/Components folder, organization users cannot publish resources, update properties, or delete resources. Organization users can run and edit reports, dashboards, Visual Analysis templates in the Public Reports folder, but cannot save the changes directly into the existing resources or into the Public Reports folder. However, they can make use of the Save As function.

Server log

Each organization has its own log files.

In the `<server_install_root>\logs` folder there are some .log files, they are for non-organization logs. For each organization there is a specific folder using the organization name and containing a set of log files for the organization.

When system admin defines detail settings about server log (log level, etc.), the settings will be applied to all organizations.

Connecting to JReport Server from JReport Designer

In JReport Designer, when publishing resources to JReport Server or synchronizing server users for CLS/RLS/MLS, you need to connect to the server firstly. In the Connect to JReport Server dialog, User Name should include the organization name when organization is enabled and when the user is a privileged organization user. Use \ to separate the organization name and the user name, for example, org1\user1. An organization user can publish resources to the My Reports/Components or Organization Reports/Components folder.

Managing tasks

Usually, a task is a set of operations you perform on JReport Server to achieve a goal, such as publishing a folder, deleting a resource, and viewing resource properties. However, the tasks that the JReport Server manages are those associated with report-running issues, the report-running tasks.

JReport Server allows you to quickly view a report result (Run), view a report result using selected options and parameters (Advanced Run), and schedule a report to run unattended at a specific time or periodically (Schedule). These are the fundamental modes that JReport Server uses to perform its report-running tasks.

You can view the status of these report-running tasks, such as scheduled tasks that are waiting to be performed by JReport Server, the tasks that are currently being performed, and the tasks that have already been performed.

This section describes the following:

- Accessing the task information tables
- Managing tasks in the task tables
- Task-level timeout for advanced run and schedule tasks

Accessing the task information tables

JReport Server collects task information and manages it in a set of tables.

- Tasks that are scheduled
  - **Scheduled table**
    Shows the status information of scheduled tasks that are waiting to be performed, such as task ID, whether the task is enabled, previous running time, and next running time.
  - **Running table**
    Shows the status information of tasks that are currently being performed, such as task ID, time when the task was started, and engine status.
  - **Completed table**
    Shows the status information of tasks that have already been performed, such as task ID, time when the task was completed, whether or not the task was successfully performed, result files, and error messages.
- Tasks that are performed in the Run, Advanced Run or Background Run mode
  - **Background Tasks table**
    Shows the status information of the tasks submitted using the Run, Advanced Run, or Background Run mode, such as report tab names, report path and name, catalog path and name, running format, time when the task was started/completed, time elapsed since the task was performed, and the status of the task.

To access a specific table, on the JReport Console page, click **My Tasks** on the system toolbar, then click the corresponding tab. The following shows the columns that are displayed in each table in detail.

Tips:

- Some columns in the tables are not shown by default. To have them displayed, switch to the table, click **Preferences** on the task bar of the My Tasks page (for the Scheduled table, click **Tools** > **Preferences** on the task bar), check the corresponding items in the Preferences dialog, then click **OK** to save the settings.
- You can make the records in each of the tables sorted ascendingly or descendingly by any column title. To do this, just click on the underlined column title on which you want the records to be sorted, you will then find a sort icon appear right after the title, showing the sort direction.

Scheduled table

The Scheduled table consists of the following columns:

Column	Description
Schedule Name	The name of the schedule task.
Task ID	The internal ID for this task, which is a unique time stamp.
Report	The report path, name and its status.
Report Tabs	The names of the report tabs in the report that are included in this task.
Next Run Time	The next scheduled time for when this task is to be performed.
Last Run Time	The last scheduled time this task was performed.
Task Type	The type of task, such as Versioning System, File System, E-mail, or Printer.
Is Enabled	Shows whether this task is enabled. Can be Enabled or Disabled.
Is Successful	Shows whether or not the last running of this task was successfully performed. The value true means that the last running was performed successfully and false means the task failed. If the column is empty, the task has not been run before.
Catalog	The catalog path and name that the report belongs to.
Launch Type	The way in which this task is executed, such as Repeatedly or One Time.
Requester	The user who submitted this task.

Running table

The Running table consists of the following columns:

Column	Description
Schedule Name	The name of the scheduled task.
Task ID	The internal ID for this task (a unique time stamp).
Report	The report path and name.
Report Tabs	The names of the report tabs in the report that are included in this task.
Start Time	The time when this task was started.
Task Type	The type of the task, such as Versioning System, File System, E-mail, or Printer.
Catalog	The catalog path and name that the report belongs to.
Launch Type	The way in which this task is executed, such as Repeatedly, One Time, or Instant.
Requester	The user who submitted this task.
Parameter File	The parameter file name. You can click the underlined file name to view the parameter values.
Parameters	The list of parameter values according to the size specified. To specify the size, click **Preferences** on the task bar, then in the Preferences dialog, set a value for the Parameter Display Size option as required.
Engine Status	The current status of the JReport engine, such as record fetching, grouping, memory paging, and engine initializing.

Completed table

The Completed table consists of the following columns:

Column	Description
Schedule Name	The name of the scheduled task.
Task ID	The internal ID for this task (a unique time stamp).
Is Successful	Shows whether this task was successfully performed. The value true means that the task was performed successfully and false that the task failed.
Report	The report path and name.
Report Tabs	The names of the report tabs in the report that are included in this task.
Completed Time	The time when this task was completed.
Task Type	The type of task, such as Versioning System, File System, E-mail, or Printer.
Catalog	The catalog path and name that the report belongs to.
Launch Type	The way in which this task is executed, such as Repeatedly, One Time, or Instant.
Requester	The user who submitted this task.
Parameter File	The parameter file name. You can click the underlined file name to view the parameter values.
Parameters	The list of parameter values according to the size specified. To specify the size, click **Preferences** on the task bar, then in the Preferences dialog, set a value for the Parameter Display Size option as required.
Engine Status	The status of JReport Engine when the task was completed, such as record fetching, grouping, memory paging, and engine initializing. When a task fails to perform, here you can see the status of the engine at the time of the error.
Error Message	The error message for when the task failed to complete the task.
Result Files	The report result file names and links to the report result files.

Background Tasks table

The Background Tasks table consists of the following columns:

Column	Description
Report Tabs	The names of the report tabs in the report that are included in this task.
Result	The result in the format in which the report ran.
Report	The path and name of the report that the report tabs belong to.
Start Time	The time when this task was started.
Finish Time	The time when this task was completed.
Status	The status of the task.
Catalog	The path and name of the catalog that the report belongs to.
Elapse Time	The time elapsed since the start of this task.
Catalog Version Number	The version number of the catalog that the report belongs to.
Report Version Number	The version number of the report.
Parameters	The parameters of the report.
Cancelled	Shows whether the task is cancelled or not.

Managing tasks in the task tables

You can manage tasks in the task tables according to your requirements. For example, you can run a scheduled task at once, or stop a running task from running.

Performing common tasks

Some task management operations are common to the task tables.

If you want to	Then do
Select a task	Click in the row that the task is in.
Select multiple tasks	Select the rows that the tasks are in while holding the Ctrl button.
Remove a task	• Select the row the task is in and click **Edit > Delete** on the task bar of the My Tasks page. • Select the row the task is in, right-click in the row and select **Delete** from the shortcut menu. • Put the mouse pointer over the row the task is in and click the **Delete** button on the floating toolbar.

Managing tasks in the Scheduled table

If you want to	Then do
Create a new scheduled task	Click **New Schedule** on the task bar of the My Tasks page, then in the New Schedule dialog, specify how to create the task: by selecting a report or by importing a script file.
Run a task at once	• Select the row the task is in and click **Run** on the task bar of the My Tasks page. • Select the row the task is in, right-click in the row and select **Run** from the shortcut menu. • Put the mouse pointer over the row the task is in and click the **Run** button on the floating toolbar.
Duplicate a task	• Select the row the task is in and click **Edit > Copy** on the task bar of the My Tasks page. • Select the row the task is in, right-click in the row and select **Copy** from the shortcut menu. • Put the mouse pointer over the row the task is in and click the **Copy** button on the floating toolbar.
Enable a task	• Select the row the task is in and click **Edit > Enable** on the task bar of the My Tasks page. • Select the row the task is in, right-click in the row and select **Enable** from the shortcut menu. • Put the mouse pointer over the row that the task is in and click the **Enable** button on the floating toolbar.
Disable a task	• Select the row the task is in and click **Edit > Disable** on the task bar of the My Tasks page. • Select the row the task is in, right-click in the row and select **Disable** from the shortcut menu. • Put the mouse pointer over the row the task is in and click the **Disable** button on the floating toolbar. The disabled task will not be performed until you enable it again.
Export a scheduled task to a script on disk	See Importing and exporting scheduled tasks for details.
Import a scheduled task from a script saved on disk	See Importing and exporting scheduled tasks for details.

Notes:

- You can perform the Run action on a disabled scheduled task.
- When copying a disabled scheduled task or exporting it to script, the disabled state will not be included since it is not a property of the task.

Managing tasks in the Running table

If you want to	Then do
Stop a task from running	• Select the row the task is in and click **Stop** on the task bar of the My Tasks page. • Select the row the task is in, right-click in the row and select **Stop** from the shortcut menu. • Put the mouse pointer over the row the task is in and click the **Stop** button on the floating toolbar. **Note:** When you stop a bursting task from running, some sub tasks in the bursting task may have already been finished, so some results may have been sent to some recipients.
View parameter information	Refer to the Parameters column of the Running table.

Managing tasks in the Completed table

If you want to	Then do
View detailed task running information	• Click the schedule name of the task. • Select the row the task is in and click **Edit** > **Details** on the task bar of the My Tasks page. • Select the row the task is in, right-click in the row and select **Details** from the shortcut menu. • Put the mouse pointer over the row the task is in and click the **Details** button on the floating toolbar.
View parameter information	Refer to the Parameters column of the Completed table.

Managing tasks in the Background Tasks table

If you want to	Then do
Stop a task submitted using Background Run mode from running	• Select the row the task is in and click **Edit > Stop** on the task bar of the My Tasks page. • Select the row the task is in, right-click in the row and select **Stop** from the shortcut menu. • Put the mouse pointer over the row the task is in and click the **Stop** button on the floating toolbar.
Restart a stopped task	• Select the row the task is in and click **Edit > Restart** on the task bar of the My Tasks page. • Select the row the task is in, right-click in the row and select **Restart** from the shortcut menu. • Put the mouse pointer over the row the task is in and click the **Restart** button on the floating toolbar.
View parameter information	Refer to the Parameters column of the Background Tasks table.

Task-level timeout for advanced run and schedule tasks

A task-level timeout mechanism is introduced in order to avoid the never-finished running tasks consuming server resources and thus decreasing the server performance. With the mechanism, you can specify a time duration for a task, and ask JReport Server to cancel the task or to notify you or someone else of the task status via e-mail if the task has not yet finished running when the task duration is up. To do this:

1. Do either of the following to enable the task-level timeout mechanism.
   - In the server.properties file, set the task.duration.enable property to **true**.
   - On the JReport Administration page, click **Configuration** on the system toolbar, select **Advanced**, and then check the **Enable Task Duration** option.
2. Set task duration check frequency. By default, JReport Server checks task duration every 30 seconds. The period value can be reset either by the task.duration.check_cycle property in the server.properties file or by the Status Refresh Interval option in the Configuration > Advanced panel on the JReport Administration page. Note that the value must be an integer greater than 0. Since task duration check frequency affects server performance, it is recommended that you set the value according to your system environment.
3. Restart JReport Server if you have modified the Advanced panel.
4. Use the Duration tab in the Advanced Run and Schedule dialogs to specify task duration.
   a. In the Timeout text box, specify the time limit for when the task can run before notifying of the timeout or canceling the task.
   b. If you want to notify someone of the task status if the task has not yet finished running when the task duration is up, check **Notify by e-mail after the specified time** and specify the mail address in the Mail To text box.
   c. If you want JReport Server to cancel the task when the task duration is up but the task is not finished yet, check **Cancel the task after the specified time**.
   d. Click **Finish** to submit the task.

Then, when the specified task duration is up but the task has not finished running,

- For an Advanced Run task,
  - If you have specified to have server cancel the task when the task duration is up, the task will be cancelled automatically, however a record for the task will be still remained in the Background Tasks table of the My Tasks page, shown with a sign of cancellation.
  - In the Duration tab, if you haven't checked the option Cancel the task after the specified time, the task will be switched to run in background mode when the task duration is up, in which case, you can choose to cancel the task manually. To do this, in the Background Tasks table,
    - Select the task row and click **Edit > Delete** on the task bar of the My Tasks page.
    - Select the task row, right-click in the row and select **Delete** from the shortcut menu.
    - Put the mouse pointer over the task row and click the **Delete** button on the floating toolbar.
- For a Schedule task,
  - If you have specified to have server cancel the task when the task duration is up, the task will be cancelled automatically, and a task completed record will be added in the Completed table of the My Tasks page, with the Is Successful status shown as No.
  - In the Duration tab, if you haven't checked the option Cancel the task after the specified time, the task will still be listed in the Running table of the My Tasks page when the task duration is up. Then, if you want to cancel the task manually, in the Running table,
    - Select the task row and click **Stop** on the task bar of the My Tasks page.
    - Select the task row, right-click in the task row and select **Stop** from the shortcut menu.
    - Put the mouse pointer over the task row and click the **Stop** button on the floating toolbar.

Notes:

- JReport Server may not cancel a task right after the specified task duration is up due to check frequency.
- It is recommended that the task duration is set to about five times of the required time for finishing running the task.

Managing server data

During its running process, JReport Server keeps track of server information and stores it to its own database for the purpose of running various managing and monitoring tasks, such as managing the resources on the server, monitoring the task running status, and collecting server running statistics.

The server database includes system, realm, and profiling. The system database holds resources of the global server scope, such as server.properties, global NLS, and so on. The realm database holds information of folders, nodes, versions, the security system, and the completed table. The profiling database holds server runtime related information. You can refer to Configuring the server database for details about how to configure the databases.

JReport provides completed SQL files to create tables for all databases supported. They reside in `<install_root>\script_files`.

This section covers the following topics:

- Initializing the database system as a non-admin database user
- Checking server data integrity
- Backing up/restoring server data
- Archiving and restoring server data
- Viewing the summary information of archive files
- Clearing unused nodes from the realm database

Initializing the database system as a non-admin database user

The database initialization system allows you to use a non-admin database user for JReport Server. However, the database initialization has to be divided into three phases:

1. The database administrator creates new database tables using script files.
2. The non-admin database user starts JReport Server. In this phase, the old version's database tables will be updated if they exist. If the server has updated the old version's table data, you will be able to see a record of it in the debug file.
3. The database administrator deletes the old version' tables in the database using script files.

Script files

JReport Server supports these different databases: Apache Derby, HSQLDB, MySQL, Microsoft SQL Server, IBM DB2, Oracle, Sybase, and Informix.

The script files are stored in `<install_root>\script_files`. There are three types of script files:

Script files used for creating new database tables

The following files are used to create new database tables, and are stored in `<install_root>\script_files\create_new_tables`:

- **db2_c.txt** - Creates new tables for DB2 database.
- **derby_c.txt** - Creates new tables for Derby database.
- **hsqldb_c.txt** - Creates new tables for HSQLDB database.
- **informix_c.txt** - Creates new tables for Informix database.
- **mysql_c.txt** - Creates new tables for MySQL database (for single-byte charset, e.g latin1).
- **mysql_mb2_c.txt** - Creates new tables for MySQL database (for MBCS two-byte charset, for example, gbk).
- **mysql_mb3_c.txt** - Creates new tables for MySQL database (for MBCS three- byte charset, for example, utf8).
- **oracle_c.txt** - Creates new tables for Oracle database.
- **sqlserver_c.txt** - Creates new tables for Microsoft SQL Server database.
- **sybase_c.txt** - Creates new tables for Sybase database.

Script files used for deleting the old version' tables

The following files are used to delete the old version' tables, and are stored in `<install_root>\script_files\delete_old_tables`:

- **db2_do.txt** - Deletes old version tables for DB2 database.
- **derby_do.txt** - Deletes old version tables for Derby database.
- **hsqldb_do.txt** - Deletes old version tables for HSQLDB database.
- **informix_do.txt** - Deletes old version tables for Informix database.
- **mysql_do.txt** - Deletes old version tables for MySQL database.
- **oracle_do.txt** - Deletes old version tables for Oracle database.
- **sqlserver_do.txt** - Deletes old version tables for Microsoft SQL Server database.
- **sybase_do.txt** - Deletes old version tables for Sybase database.

Script files used for deleting the current version' tables

The following files are used to delete the current version' tables, and are stored in `<install_root>\script_files\delete_current_tables`:

- **db2_dc.txt** - Deletes current version tables for DB2 database.
- **derby_dc.txt** - Deletes current version tables for Derby database.
- **hsqldb_dc.txt** - Deletes current version tables for HSQLDB database.
- **informix_dc.txt** - Deletes current version tables for Informix database.
- **mysql_dc.txt** - Deletes current version tables for MySQL database.
- **oracle_dc.txt** - Deletes current version tables for Oracle database.
- **sqlserver_dc.txt** - Deletes current version tables for Microsoft SQL Server database.
- **sybase_dc.txt** - Deletes current version tables for Sybase database.

Checking server data integrity

In abnormal circumstances, server data may not be saved correctly or completely in the databases. JReport Server allows you to check the integrity of server data.

The integrity check mainly examines two aspects of the realm database:

- Integrality and consistency among tables - checking records among the tables to see whether they are complete and consistent.
- Integrality and consistency between realm database and the external files - checking to see whether the records in the tables match the related external files.

If any inconsistent or incomplete server data is found, it will be removed from the realm database since it is unused. The same process also applies to files.

To check server data integrity, do one of the following:

- Launch JReport Server on the command line, using parameter `-cleanup`.

  ```
  C:\JReport\Server\bin>JRServer -cleanup
  ```

- Open the batch file used to start JReport Server, add **`-Dcheck.realmtables=true`** to the line that starts JReport Server:

  ```
  "%JAVAHOME%\bin\java.exe" -Dcheck.realmtables=true "-Dinstall.root= <install_root>" ...
  ```

Note: The integrity check checks only the realm database.

Backing up/restoring server data

Backing up server data

The server data backup process inspects all of the tables in the database, collecting and then exporting all table data to a ZipEntry. For certain tables, such as the catalog version table, report version table, and report result version table, the data also includes relational real files from the history directory. All server data is compressed to a single Zip file, and all table data and every relational real file is stored as a ZipEntry in this single file.

You have two alternatives for exporting the server data: through the server user interface or through command line.

Exporting the server data through the server user interface

1. On the JReport Administration page, click **Data** on the system toolbar and then select **System DB**/**Realm DB**/**Profiling DB** from the drop-down menu.
2. Select a realm if it is the Realm DB or Profiling DB panel.
3. Click the **Backup** tab, type the file path and name in the Backup System DB/Realm DB/Profiling DB field (or use the Browse button to specify the file path), and then click **Backup**. Note that the file extension should be included.

Exporting the server data through the command line on Windows or on Unix/Linux

A tool is provided to backup the server data. It is DBMaintain.bat/DBMaintian.sh in `<install_root>\bin`. It has two parameters:

- -Bsystemtables/-Brealmtables/-Bprofiling: Backup the data with the server data in a JReport cluster.
- -B0realmtables: Only backup the data in the database.

To export the server data from the command line on Windows or on Unix/Linux:

1. In a DOS window, switch to the `<install_root>\bin` folder.
2. Use the `DBMaintain` command (DBMaintain.sh on Unix/Linux) and -Bsystemtables/-Brealmtables/-Bprofiling or -B0realmtables parameter. For example:

   `C:\JReport\Server\bin>`**`DBMaintain -Brealmtables:`**`C:\TEMP\cmd_b_realmtables.dat` **`-Bprofiling:`**`C:\TEMP\cmd_b_profiling.dat`

Restoring server data

The server data restore process picks up all table data from the backup file and inserts it into the corresponding table in the database. You can only restore server data using the DBMaintain tool with the -Rsystemtables/-Rrealmtables/-Rprofiling or -R0realmtables parameter.

To import server data using the command line:

1. In the DOS window, switch to the `<install_root>\bin` folder.
2. Use the DBMaintain command and -Rsystemtables/-Rrealmtables/-Rprofiling or -R0realmtables parameter, for example:

   `C:\JReport\Server\bin>`**`DBMaintain -Rrealmtables:`**`C:\TEMP\cmd_b_realmtables.dat` **`-Rprofiling:`**`C:\TEMP\cmd_b_profiling.dat`

Backup/restoration limitations

The backup/restore feature does not support cross-platform operation. The backup and restore operations must be done on the same operating system. For example, if you backup the server data to a zip file on a Windows platform, you will then not be able to restore it on a Unix system.

Notes:

- On the JReport Administration page, the Profiling DB option is not shown by default on the Data drop-down menu. To make it shown, you need to set the server.profiling.enable property to true in the server.properties file in the `<install_root>\bin` directory.
- When running DBMaintain.bat/DBMaintian.sh, error may occur if the backup zip file is too large, which is caused by JVM limitation. You can try the -B0realmtables and -R0realmtables options to backup and restore data separately.
- When backing up server data in cluster environment, it is recommended that you back up the data on every cluster node to make sure no files get lost on any node after they are restored. Also, the system DB and realm DB should better be backed up together. Since the external data cannot be backed up once just on one single cluster node, to avoid redundant data backup, you need to manually back up the external files on every cluster node after backing up the system DB and realm DB on any of the cluster nodes.
- When restoring server data in cluster environment, first make sure all the cluster nodes are shut down, and then restore the data on each cluster node.

Archiving and restoring server data

You can archive and restore server data in the realm and profiling databases.

Archiving server data

Along with the running of the server, the sizes of the result version table (DB: realm) and TaskContext table (DB: profiling) grow larger. JReport Server allows you to archive data to a single backup file. You can then retrieve the data from the backup file at a later date.

To archive the server data:

1. On the JReport Administration page, click **Data** on the system toolbar and then select **Realm DB**/ **Profiling DB** from the drop-down menu.
2. Select a realm from the Select Realm drop-down list.
3. Switch to the **Archive** tab.
4. Specify a date in the Move to Archive Data Before field, and then type or use the **Browse** button to specify the file path and name.

   If you don't want to backup the archived server data, leave this field blank. The server data will then be removed from the database.
5. Click **Archive**. Server data prior to the date specified will be archived and saved to the backup file using the name specified.

Restoring the archived server data

You can restore server data that has been archived from a backup file. You can only restore it using the server user interface.

To restore the archived server data:

1. On the JReport Administration page, click **Data** on the system toolbar and then select **Realm DB**/ **Profiling DB** from the drop-down menu.
2. Select a realm from the Select Realm drop-down list.
3. Switch to the **Archive** tab.
4. Type or use the **Browse** button to provide the archive file name and location in the Restore from archive field, and then click **Restore**.

Notes:

- The Profiling DB option is not shown by default on the drop-down menu. To make it shown, set the server.profiling.enable property to true in the server.properties file in the `<install_root>\bin` directory.
- You can only restore server data from the backup file that is archived using the same server version.

Viewing the summary information of archive files

After archiving or backing up files, you can view the results. The summary information includes Archive, Date, Type, Version, Realm, Database and Scope.

To view the summary information of a specified file:

1. On the JReport Administration page, click **Data** on the system toolbar and then select **Realm DB**/ **Profiling DB** from the drop-down menu.

   Note: The Profiling DB option is not shown by default on the drop-down menu. To make it shown, set the server.profiling.enable property to true in the server.properties file in the `<install_root>\bin` directory.

2. Select a realm from the Select Realm drop-down list.
3. Switch to the **Summary** tab.
4. Type or use the **Browse** button to specify an archive file name and location in the Archive field, and then click **Summary**.

Clearing unused nodes from the realm database

Making operations on real path resources may generate unused nodes from the realmdatabase. The generated unused nodes can be cleared through the JReport Administration page.

How unused nodes are generated

When resources, such as folders, catalogs, and reports, have been published using a real path in order to allow dynamic updating there is a possibility to produce invalid nodes. After running a report and saving the result to the versioning system or scheduling reports which are saved in the versioning system, either of the following conditions will generate invalid nodes:

- Cancel or change the real path setting.
- Publish a resource to replace an existing real path resource.

How to clear invalid nodes

To clear unused nodes:

1. On the JReport Administration page, click **Data** on the system toolbar and then select **Realm DB** from the drop-down menu.
2. Click the **Realm DB Check** tab.
3. Click the **Check** button. All invalid resource nodes will then be listed.
4. Check the checkbox on the header of the first column to specify whether you want to select all or unselect all. Otherwise, you can check the corresponding checkbox to select the invalid nodes that you want to delete.
5. Click **Delete** button to delete all selected nodes.

Managing cached report data

Cached report data (CRD) is a cached subset of data fetched from the database according to certain conditions and is used as a substitute for the database for retrieving data for reports. This provides several benefits.

- Reports can be run from a specific point in time such as a month end report or quarter end report without going back to the DBMS to get the original data.
- Many users can run reports from data created from a single query without impacting production DBMS users.
- Users running a report will all see the same view of the data, thus the data will not change minute by minute based on current DBMS updates.
- Caching report data can be scheduled for any time frame to simulate real time on-demand reporting for many users while not slowing down the production DBMS.

Cached report data can be created for data sources including queries, stored procedures, imported SQLs, user defined data sources, hierarchical data sources, and parameters whose type is Bind with Single Column or Bind with Cascading Columns. All these types of data sources will be called query resources for convenient documentation since they will be mentioned a lot in the following sections.

There are two types of cached report data in JReport Server: auto CRD and scheduled CRD.

Auto CRD

When running a report, if there is no scheduled CRD created for the query resource that the report is using directly or indirectly (for example, the report is created based on a business view and the business view is built from a query resource), data will be fetched from the data source, and at the same time the fetched data is cached and becomes an auto CRD.

When there are auto CRDs generated in a query resource, the report running request based on the query resource will first search for the auto CRD that contains all the data required by the report. If it finds one, the located CRD will retrieve data to the report, but if JReport cannot find one, data is fetched from the data source and cached to be another auto CRD.

Auto CRDs will only be available within one server running life cycle, which means that once the server closes or restarts, they will be removed and a new cycle of auto CRD generation will begin. How many auto CRDs can be held in a query resource within one server running session is decided by the maximum hard disk space configured in the Cache Configuration dialog.

Auto CRDs are disabled for generation by default. You can enable them and configure the maximum hard disk space for auto CRDs and how long an auto CRD is stored. For details, see the Automatic Cache section in the Cache Configuration dialog.

Scheduled CRD

Scheduled CRDs can be created and managed by server administrators. A query resource can have zero or one scheduled CRD. When a query resource contains parameters, its scheduled CRD can only represent the data of one parameter scenario.

Via JReport's scheduling mechanism, administrators are able to define when a scheduled CRD for a query resource will be created and how it will be updated according to time. Scheduled CRDs have no version: once they are updated, only the latest are kept in the query resource.

Creating scheduled CRDs

Administrators can schedule to have a data cache created for a query resource and updated at a specific time or periodically. When scheduling a CRD task, more than one query resource can be selected at a time and a CRD can be created for each query resource and all of the scheduled CRDs will be applied the same creation and updating policy.

1. On the JReport Administration page, click **Cache** on the system toolbar and then select **Data Cache** from the drop-down menu.
2. On the Cached Report Data tab, click **New Cache** on the task bar.
3. In the New Cache dialog for selecting queries, click ... next to the Select a Folder text field.
4. In the Select Folder dialog, browse to the folder containing the required catalog and click **OK**.
5. From the Select a Catalog drop-down list, select the required catalog in the specified folder.
6. Click + beside the **Select Queries** box.
7. In the Select Queries dialog, select the query resources you want to create data caches for from the resource tree. Then click **OK**.
8. Click **OK** in the New Cache dialog for selecting queries and you are then redirected to the New Cache dialog for scheduling.
9. In the General tab, you can specify values to the parameters of the query resources listed in the Select Query drop-down list. If no values are provided, the default values will be applied. The query resource names will be used as the new CRD names.
10. Specify the other settings in the tab if needed.
11. In the Conditions tab, specify the time for when the task is to be performed in the Time sub tab, and select or create a trigger to bind with the task in the Trigger sub tab. The setting in the Conditions tab decides when the CRDs will be created and when they will be updated.
12. In the Notification tab, specify to notify someone via e-mail when the task is finished and whether it is successful or unsuccessful.
13. Click **Finish**, and JReport Server will then perform the task at the requested times.

Using scheduled CRDs

A scheduled CRD has four statuses. Whether reports running based on the same query resource as the CRD can fetch data from the CRD is determined by the CRD's status.

- **Not ready**
  CRD is not initiated yet. The reports will try to get data from the data source directly.
- **Initiating**
  CRD is preparing its data for the first time, but still not ready. The reports will wait for the CRD to be ready and then fetch data from the ready CRD.
- **Ready**
  CRD is prepared and ready to use. All reports based on the same query resource will fetch data from the CRD.
- **Updating**
  CRD is updating itself at a scheduled point. All reports based on the same query resource will use the ready CRD before the update happens. Once the update is finished, later reports will get data from the updated CRD.

After a scheduled CRD is created and ready to use, all reports based on the same query resource as the CRD will automatically use the CRD for retrieving data.

Since a scheduled CRD freezes parameters in the query resource, if a report uses a scheduled CRD and its query resource contains parameters, when running the report, only parameters used in the report are available for specifying values and the query resource parameters will be disabled for editing. In JReport, a parameter may depend on another parameter; if the latter is frozen, the former will be frozen as well.

Editing scheduled CRDs

The generated scheduled CRDs are displayed in the Cached Report Data tab.

You can further edit the scheduling information of the CRDs if required. To do this, select the row in which the CRD you want to edit is located, then click **Edit > Properties** on the task bar or right-click on the row and select **Properties** from the shortcut menu. If parameters or schedule policy is changed, they will only take effect after the next CRD update; before that, all reports using the CRD will still get the old data.

In addition, if you find any of the scheduled CRD is no longer required, you can remove it. To do this, select the row where the CRD is, then clicking **Edit > Delete** on the task bar; or right-click in the row and select **Delete** from the shortcut menu.

Viewing and managing scheduled CRD tasks

CRD tasks that are waiting to be performed are listed in the Scheduled tab. To access the tab, click **Cache** > **Data Cache** on the system toolbar and then go to the **Scheduled** tab.

CRD tasks that are currently being performed are listed in the Running tab. To access the tab, click **Cache** > **Data Cache** on the system toolbar and then go to the **Running** tab.

CRD tasks that have already been performed are listed in the Completed tab. To access the tab, click **Cache** > **Data Cache** on the system toolbar and then go to the **Completed** tab. You can remove the record of a completed CRD task from the tab if required. To do this, select the row where the record is located and then click **Delete** on the task bar, or right-click on the row and select **Delete** from the shortcut menu.

For details about the tabs, refer to Data Cache panel.

Recovering scheduled CRDs

Scheduled CRDs will be recovered after the server shuts down and then restarts. The data before the shutdown will be maintained.

If a CRD is updating when the shutdown happens, the CRD will continue with the update when the server restarts.

Synchronizing scheduled CRDs with catalog republish

After a catalog is republished, how the existing scheduled CRDs based on the old query resources in the catalog will be updated depends on the following conditions:

- If no query resource is modified, the existing CRDs will be automatically applicable to the new catalog.
- If a query resource is modified, the scheduled CRD based on the old query resource will behave as follows:
  - Before the next schedule time, the CRD will be disabled for retrieving data to reports.
  - When it is the next schedule time, data is fetched from the new query resource to generate an updated scheduled CRD:
    - If new parameters are added in the new query resource, default values will be used until an administrator specifies values to them.
    - If old parameters are deleted from the new query resource, the parameter values previously specified in the CRD will be removed.
    - If parameters are changed and the previously specified values do not match the type, error messages will be given in the server error log and the CRD is disabled for service.

Configuring the CRD memory usage

Administrators can configure the maximum CRD memory in the Cache Configuration dialog, which is applicable for both types of CRDs (to access the dialog, focus on the Cached Report Data tab, then click **Cache Configuration** on the task bar). The value should be at least 4 MB and not more than 80% of the JVM current maximum heap size. The number of MBs must be configured in whole numbers. The cache is used to improve performance when multiple users are running reports using the same CRD. Any size CRD can be accessed with even the smallest cache size. The larger the cache size the faster performance will be for concurrent users, but a large cache size may lower performance for users who do not use the CRD. All CRDs share the same cache area.

If possible, set the maximum memory usage to a number that the memory can handle, for example, 512 MB or bigger.

Managing in-memory cubes

In-memory cubes are cached data of business views. The generating and updating of in-memory cubes are achieved by scheduling cube tasks. In-memory cubes are similar to scheduled reports that can be run immediately, run once at a specific time or run periodically to select the data from the DBMS and store it in JReport Server. After an in-memory cube is created and ready for use, all web reports, library components, and visual analyses that use the same business view as the in-memory cube will automatically use the cube rather than go to the database for the data. By applying in-memory cubes, your reporting performance will be greatly improved.

This section shows how to create, use and manage in-memory cubes as follows:

- Creating in-memory cubes
- Viewing and managing the cube tasks
- Recovering and synchronizing in-memory cubes

Creating in-memory cubes

A business view can have only one in-memory cube. When a business view contains parameters, its in-memory cube can only represent the data of one parameter scenario. However, if you want the in-memory cube to be available to Visual Analysis, the cube should contain all of the parameters in the business view.

Via JReport's scheduling mechanism, administrators are able to define when an in-memory cube for a business view will be created and when it will be updated at a specific time or periodically. In-memory cubes have no versions: once they are updated, only the latest one is kept for the business view.

If it is the first time to create in-memory cubes, the administrator needs to first configure the maximum memory allowed for all in-memory cubes.

To configure the maximum memory for in-memory cubes:

1. On the JReport Administration page, click **Cube** on the system toolbar.
2. In the Cube panel, click **Configuration** on the task bar.
3. In the Cube Configuration dialog, specify a value no larger than 80% of your Java Heap size for the Maximum Cube Memory Allowed option.
4. Click **OK**.

It can also be specified by the property server.cube.maxmemory in the server.properties file in the `<install_root>\bin` directory.

To create an in-memory cube:

1. On the JReport Administration page, click **Cube** on the system toolbar.
2. In the Cube panel, click **New Cube** on the task bar.
3. In the New Cube dialog for selecting business view, click ... next to the Select a Folder text field.
4. In the Select Folder dialog, browse to the folder containing the required catalog and click **OK**.
5. From the Select a Catalog drop-down list, select the required catalog in the specified folder.
6. In the Select Business View box, select the business view on which to create the in-memory cube.
7. Click **OK** and you are then directed to the New Cube dialog for scheduling.
8. In the General tab, you can specify values to the parameters of the business view if there are any. If no values are provided, the default values will be applied. The business view name will be used as the new in-memory cube name. Specify the other settings in the tab if needed.
9. In the Schedule tab, specify the time for when the task is to be performed in the Time sub tab, and select or create a trigger to bind with the task in the Trigger sub tab. The setting in the Schedule tab decides when the in-memory cube will be created and when it will be updated.
10. In the Settings tab,
    a. In the Maximum Memory Allowed text box, provide a value for the cube. The value must be specified, otherwise you are not allowed to continue with other tabs or to finish the dialog.
    b. If you would like to cache the cube on disk once the available memory is exhausted, select **Move the cube to disk if insufficient memory was allocated**. If this option is not selected and the memory is not enough for the cube, the cube status will be disabled and an error will be shown in the log tab.
    c. If you would like the detail data in the business view to be cached onto disk, select **Cache Detail on Disk**. Otherwise if you need only the aggregation data to be cached, unselect the option.
11. In the Notification tab, specify to notify someone via e-mail when the task is finished and whether it is successful or unsuccessful.
12. Click **Finish**, and JReport Server will then perform the task at the requested times.

After an in-memory cube is created and ready for use, all web reports, library components and visual analyses based on the same business view as the cube will automatically use the cube for retrieving data.

Since an in-memory cube freezes parameters in its related business view, if a report uses an in-memory cube and its business view contains parameters, when running the report, only parameters used in the report are available for specifying values and the business view parameters will be disabled for editing. In JReport, a parameter may depend on another parameter; if the latter is frozen, the former will be frozen as well.

Viewing and managing the cube tasks

You can view information of the tasks scheduled for creating in-memory cubes and manage the tasks in the following tabs.

Schedule tab

All scheduled cube tasks are listed in the Schedule tab (to access the tab, click **Cube** on the system toolbar and then go to the **Schedule** tab), which consists of the following columns:

Column	Description
Name	The name of the cube task.
Catalog	The catalog that the cube belongs to.
Data Source	The data source that the cube belongs to.
Status	The task status: • **Pending** The cube has no data and is waiting to run based on scheduled policy. The cube is not ready for use. • **Building** It is the first time creating the cube and filling data to the cube. The cube is not ready for use. • **Updating** It is updating cube data based on the scheduled policy. The cube is not ready for use. • **Completed** Cube data is available. The task will not be updated any more according to the scheduled policy. • **Disabled** The cube is not allowed to generate or update following the scheduled policy. The cube data cannot be used or released.
Create Time	The time when the scheduled task was created.
Next Run Time	The next time when the scheduled task will generate or update the cube.
Last Run Time	The last time when the scheduled task generated or updated the cube.
Times Run	The number of times the scheduled task has generated or updated the cube. If the number is bigger than 0, click it and you will be able to view the details of each running of the cube, including all updating events since the cube was generated. The detail table contains the following columns: **Column** / **Description** Activity: Shows the type of the event. • **Build** Indicates that the cube was created. • **Update** Indicates that the cube was updated. Start Time: Shows the time when the event started. End Time: Shows the time when the event ended. Status: Shows whether the event is successful. • **Successful** Indicates that the event is successful. • **Failed** Indicates that the event failed.

You can edit any of the cube tasks in the Schedule tab in the following ways:

- To further edit the scheduling information of a task, select the row in which the task is located, then click **Edit > Properties** on the task bar, or the **Properties** button on the floating toolbar, or right-click on the row and select Properties on the shortcut menu. In the displayed dialog, edit the scheduling information. If parameters or schedule policy is changed, they will only take effect after the next cube update; before that, all reports using the cube will still get the old data.
- To disable a task that is not disabled, select the row where the cube task is, then click **Edit > Disable** on the task bar, or the **Disable** button on the floating toolbar, or right-click on the row and select **Disable** on the shortcut menu.
- To enable a task that is disabled, select the row where the cube task is, then click **Edit > Enable** on the task bar, or the **Enable** button on the floating toolbar, or right-click on the row and select **Enable** on the shortcut menu.
- To remove a task that is no longer required, select the row where the cube task is, then click **Edit > Delete** on the task bar, or the **Delete** button on the floating toolbar, or right-click on the row and select **Delete** on the shortcut menu.

Active tab

All the cube tasks that have generated in-memory cubes which have data and are ready for use are listed in the Active tab (to access the tab, click **Cube** on the system toolbar and then go to the **Active** tab), which consists of the following columns:

Column	Description
Name	The name of the cube task.
Catalog	The catalog that the cube belongs to.
Data Source	The data source that the cube belongs to.
Updated Time	The last time when the cube was updated.
Allocated	The maximum memory allowed for the cube.
Actual	The memory size currently used by the cube.
Located At	Shows the location of the cube: • **On Disk** Indicates that the cube is swapped to the disk. • **In Memory** Indicates that the cube is in the memory.
Parameter	Displays a parameter icon. By clicking the icon a dialog will appear to show all the parameters used in the cube.
Usage	The number of reports and library components that have accessed the cube. If the number is bigger than 0, click it and you will get a detail table showing the information of all the web reports and library components that used data from the cube. The detail table contains the following columns: **Column** / **Description** Name: The name of the web report or the library component that used the cube. Type: Indicates whether it is a web report or a library component. Dashboard: The name of the dashboard that is using the library component. For library components only. User: The user who ran the web report or the library component. Description: The description about the web report or the library component.

In the Active tab, you can also edit the tasks in the same way as you do in the Schedule tab.

Log tab

All the events happened on all the cube tasks are recorded in the Log tab (to access the tab, click **Cube** on the system toolbar and then go to the **Log** tab), which consists of the following columns. The number of the records allowed in the tab is controlled by the property "server.cube_log.max_count". The default number is 3000.

Column	Description
Name	The name of the cube task.
Catalog	The catalog that the cube belongs to.
Data Source	The data source that the cube belongs to.
Activities	The type of the event: • **Create** The cube was created. • **Edit** The cube was edited. • **Initiate** The cube data was initiated and made usable. • **Update** The cube data was updated. • **Disable** The cube was disabled. • **Enable** The cube was enabled. • **Delete** The cube was deleted.
Start Time	The time when the event started.
End Time	The time when the event ended.
Status	The event status: • **Successful** The event was successful. • **Failed** The event failed.
Message	Messages for failed events.

You can delete any of the log records from the tab if you want. To do this, select a record row and then click **Edit > Delete** on the task bar, or the **Delete** button on the floating toolbar, or right-click on the record row and select **Delete** on the shortcut menu.

Tip: Some columns are not shown by default in the three tabs. To have them displayed, focus on the required tab, click **Preferences** on the toolbar, check the corresponding items in the Preferences dialog, and then click **OK** to apply the settings.

Recovering and synchronizing in-memory cubes

Recovering in-memory cubes

If an in-memory cube is updating when the server shuts down, the cube will be disabled after the server restarts. Administrators need to enable the cube task in order for the cube to continue to update at the next scheduled time.

Synchronizing in-memory cubes with catalog republish

After a catalog is republished, how the existing in-memory cubes based on the old business views in the catalog will be updated depends on the following conditions:

If no business view is modified, the existing in-memory cubes will be automatically applicable to the new catalog.

If a business view is modified, the in-memory cube based on the old business view will behave as follows:

- Before the next schedule time, the cube will be disabled for retrieving data to reports.
- When it is the next schedule time, data is fetched from the new business view to generate an updated cube:
  - If new parameters are added in the new business view, default values will be used until an administrator specifies values to them.
  - If old parameters are deleted from the new business view, the parameter values previously specified in the cube will be removed.
  - If parameters are changed and the previously specified values do not match the type, error messages will be given in the server error log and the cube is disabled for service.

National Language Support

JReport products implement a National Language Support (NLS) feature, with which you can run JReport Server (including Page Report Studio, Web Report Studio, JDashboard and Visual Analysis), and reports in different language environments.

NLS can also be edited on JReport Designer.

This chapter is split into two topics:

- NLS at application level
- NLS at report level

Note: NLS feature is not supported on shape map.

NLS at application level

In JReport Server, a folder "resources" is added in the installation root directory for holding language resources. The resources are classified into two types, language packages and resources for JReport Engine. To make JReport Server displayed in your desired language completely, you need to create the language files for these two types of resources respectively.

When creating a WAR/EAR file to include a JReport Server, the languages.jar which packs all language resources in the `<server_install_root>\resources` directory will be generated and included for the multiple language support. The languages.jar makes sure the server UI text is displayed correctly after deploying the WAR/EAR to other application servers.

Creating language packages

The language packages hold almost all the UI text and messages available in JReport Server (including Page Report Studio, Web Report Studio, JDashboard and Visual Analysis). When a language package is applied, the UI will be displayed in the language specified in the language package. When there are more than one language packages, you can select the one you are familiar with for your own convenience as the JReport Server environment language. For example, if you are a German you may be glad to apply the German language package.

Currently JReport provides only the English language packages "en" and "en-us" for the English version and the Chinese language package "zh-cn" for the Chinese version. However, JReport accepts user-customized language packages and can recognize and load them if they are correctly specified.

In each language package, there are three property files which together contain all UI text and messages in JReport Server:

- **common.properties**
  This property file stores some common UI text all over JReport Server (including Page Report Studio, Web Report Studio, JDashboard and Visual Analysis) in the specific language.
- **dhtml.properties**
  This property file stores UI text and messages referred by Page Report Studio, Web Report Studio, JDashboard and Visual Analysis in the specific language.
- **server.properties**
  This property file stores UI text and messages referred by JReport Server in the specific language.

The structure of a language package which is stored in the `<server_install_root>\resources\server\languages` directory follows:

```
\LanguageName
        \properties
                common.properties
                dhtml.properties
                server.properties
```

Manually adding a language package

To add a language package, follow the steps below:

1. Browse to the `<server_install_root>\resources\server\languages` directory.
2. Create a folder named fr. The folder name should keep to the naming criterion.
3. Copy the properties folder in the existing `<server_install_root>\resources\server\languages\en` directory to the \fr folder.
4. Modify the three property files: common.properties, dhtml.properties, and server.properties in the \fr folder by translating all the text and messages after "=" to French.
5. Save these property files with UTF-8 encoding.
6. Convert the contents in the three property files into Unicode using native2ascii.exe in the `<jdk_install_root>\bin` directory by running the following line in the Command Console:

   ```
   C:\jdk1.6.0_17\bin>native2ascii -encoding utf-8 common.properties >newcommon.properties
   C:\jdk1.6.0_17\bin>native2ascii -encoding utf-8 dhtml.properties >newdhtml.properties
   C:\jdk1.6.0_17\bin>native2ascii -encoding utf-8 server.properties >newserver.properties
   ```

   Note: When you convert your property files to the same directory as the original ones you need give them new names instead of replacing the original in order to avoid problems.
7. Rename the original property files, you may want to modify them more at a later time.
8. Change the names of the generated property files back to the same names as the original property files: newcommon.properties to common.properties, newdhtml.properties to dhtml.properties, and newserver.properties to server.properties.
9. Restart JReport Server.

Naming criterion for language package folders

* FolderName(language)

* FolderName(language-country)

* FolderName(language-country-variant)

The folder name should be lower-case codes.

The language argument is a valid ISO Language Code as defined by ISO-639. You can find a full list of these codes at a number of sites, for example: http://www.loc.gov/standards/iso639-2/php/code_list.php.

The country argument is a valid ISO Country Code as defined by ISO-3166. You can find a full list of these codes at a number of sites, for example: http://www.chemie.fu-berlin.de/diverse/doc/ISO_3166.html.

The variant argument is a vendor or browser-specific code. For example, use win for Windows, mac for Macintosh, and posix for POSIX. Where there are two variants, separate them with an underscore, and put the more important one first.

Specifying the default language package

When the language packages are created, you can specify which of them will be applied to your server. There are two ways to specify the default language package:

- **Specifying by UI option**

  There is an option Specify Default Language for you to switch the language. The available language list depends on the language packages in `<server_install_root>\resources\server\languages`.

  To specify the language:

  - For administrators, on the JReport Administration page, click **Profile** on the system toolbar, select **Customize Server Preferences** from the drop-down menu, then click the **Advanced** tab and set the default language for all users.
  - For end users, on the JReport Console page, click **Profile** on the system toolbar, then click **Customize Server Preferences** on the task bar of the Profile page, select the **Advanced** tab and set a language for themselves.

- **Specifying by URL parameter**

  End users can control the language by the parameter jrs.language when accessing the JReport Console page or Page Report Studio UI via URLs. The format of the URL is as follows:

  ```
  http://IP_Address:port/context/pagename?parameter=value&jrs.language=Language_Name
  ```

  The value of jrs.language should be the same as the language package folder name in the `<server_install_root>\resources\server\languages` directory and be lower-case letters.

  The following are two examples:

  - URL for accessing the JReport Console page:

    ```
    http://127.0.0.1:8888/jinfonet/index.jsp?jrs.language=zh-cn
    ```

  - URL for opening a page report in Page Report Studio:

    ```
    http://127.0.0.1:8888/dhtmljsp/dhtml.jsp?jrs.cmd=jrs.try_vw&jrs.report=/SampleReports/Employee Information.cls&jrs.catalog=/SampleReports/SampleReports.cat&jrs.cat_version=1&jrs.path=/SampleReports/Employee Information.cls&jrs.result_type=8&jrs.language=zh-cn
    ```

Note: The specified language by URL parameter has higher priority than that specified by UI option; however, it takes effect only in the current user session.

Creating the JReport Engine language resource files

The JReport Engine language resources are saved in the following files in the `<server_install_root>\resources\common\resource` directory: JRError.properties, JRMessage.properties, JVMessage.properties and JVMisc.properties. To create the files in a specified language, follow the steps below:

1. Save the files with new names which contain the ISO language and country code suffix in the same directory (for more information, see the above naming criterion). For example, if you want to create the French language files, save them as JRError_fr_FR.properties, JRMessage_fr_FR.properties, JVMessage_fr_FR.properties and JVMisc_fr_FR.properties.
2. Translate all the text after "=" in the newly saved files to French.
3. Save the files with UTF-8 encoding.
4. Convert the contents in the files into Unicode and rename them to the original ones with the language and country suffix.
5. Restart JReport Server.

Which language resource files will be applied depends on the language and locale settings of your system.

NLS at report level

If the NLS feature is enabled for a report or library compnent when it is designed in JReport Designer, it will also be available after the report or library compnent has been published to JReport Server. Then, when the report or dashboard that contains the library compnent runs in the client/server scenario, different clients can select different languages to render it. Also, JReport provides you with the NLS Editor on JReport Server, with which you can edit NLS for any report or library compnent the same as the report designer does in JReport Designer, and what's more, you can create global NLS resources that can be shared by all reports and library compnents to reduce the translation cost.

The following topics show how to use the NLS feature in JReport Server:

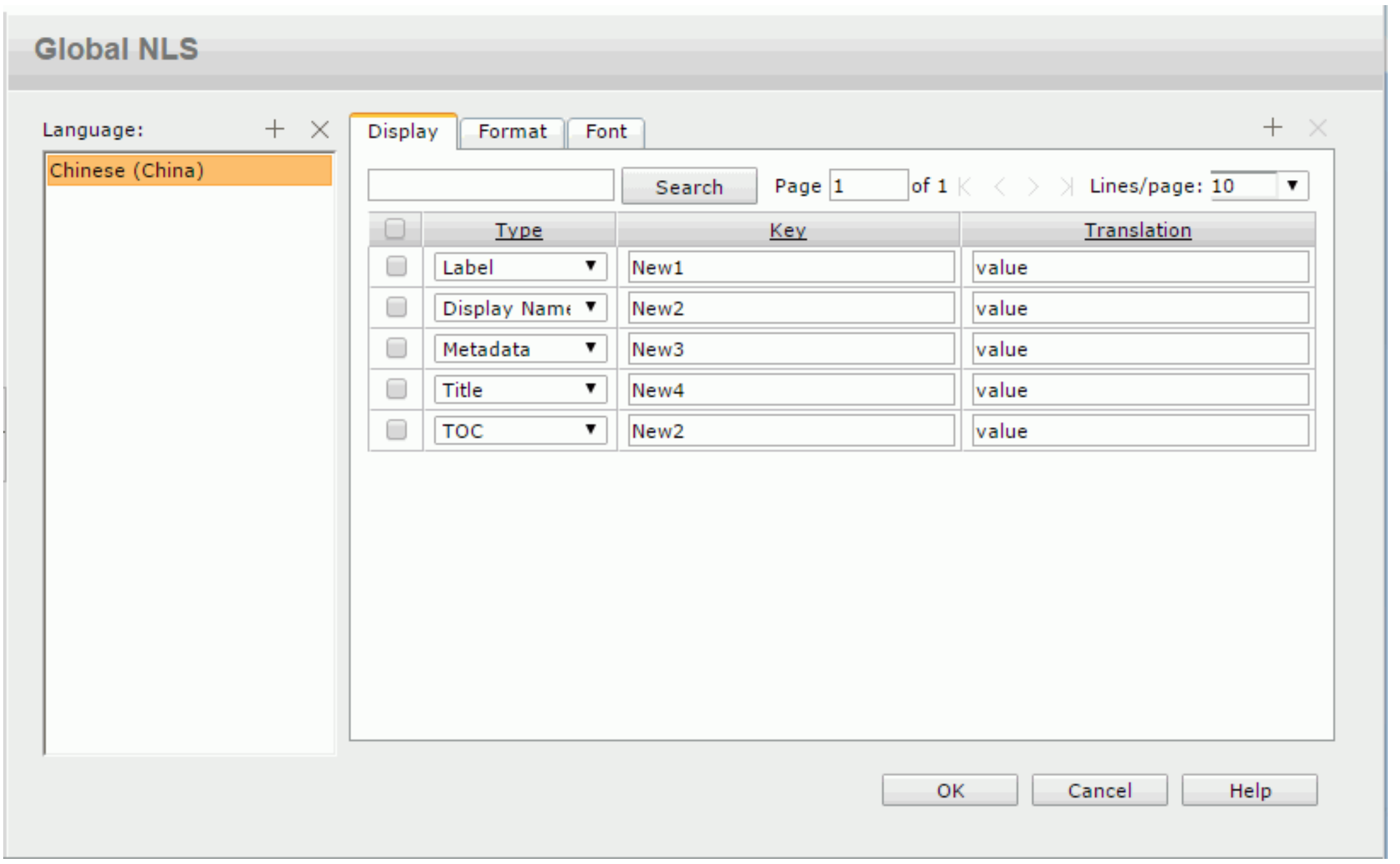
- Creating global NLS
- Editing local NLS
- Editing resource tree NLS
- Running NLS reports/dashboards
- Localizing page navigation links in HTML report outputs

Creating global NLS

The global NLS is similar to the local NLS feature. Local NLS is the NLS resource used for a particular report or library component, while global NLS is the NLS resource that can be used for all reports and library components in any catalog in both JReport Server and JReport Designer. With global NLS, you can share the NLS information between all reports and library components and reduce your translation cost. This feature is provided to administrators only.

To create global NLS on JReport Server:

1. Start JReport Server and log onto the JReport Administration page.
2. Click **Resources** on the system toolbar and select **Global NLS** from the drop-down menu.



3. Click the **Add** button + above the Language box. The Select Language Source dialog is then displayed.
4. Specify where to add the required languages.
   - **Languages Supported by JReport**
     Adds languages from the languages that JReport supports.
   - **NLS Resource File**
     Adds languages from an external NLS resource file which has been defined with some language information.

     NLS resource files should follow the naming rule: NLS_[language]_[region A2]_[User Defined].properties. The language argument is a valid ISO Language Code as defined by ISO-639. You can find a full list of these codes at a number of sites, for example: http://www.loc.gov/standards/iso639-2/php/code_list.php. The region argument is a valid ISO Country Code as defined by ISO-3166. You can find a full list of these codes at a number of sites, for example: http://www.chemie.fu-berlin.de/diverse/doc/ISO_3166.html.
5. Click **OK** in the Select Language Source dialog. Then,
   - If Languages Supported by JReport is checked in the dialog, the Add Language dialog will be displayed. Select the required languages and click **OK** to add them.
   - If NLS Resource File is checked in the dialog, the File Upload dialog will be displayed. Browse to the local folder where the NLS resource file is located, select the file and then click **Open**.

   The specified languages are now displayed in the Language box in the Global NLS dialog.
6. Select a language from the Language box to edit global NLS for it.
7. In the Display tab, click the **Add** button + to add a new row of display. Select the type of the display from the Type drop-down list, specify the key in the Key column, then give the corresponding target language text in the Translation column. Click + to add more rows and specify the key and translation according to your requirements.
8. In the Format tab, click the **Add** button + to add a new row of format. In the Key column, specify the format of the key in the original language, then in the Format column, specify the format of the key in the target language. Click + to add more rows and specify the format information according to your requirements.
9. In the Font tab, click the **Add** button + to add a new row of font. In the Key column, choose from the drop-down lists the font face and font size of the key, in the Font Face column, choose from the drop-down list the font face for the target language, then in the Font Size column, choose from the drop-down list the font size for the target language or check to use relative font size. Click + to add more rows and specify the font information according to your requirements.
10. Repeat the above steps to define global NLS for the other languages as required.
11. When done, click **OK** to accept the settings.

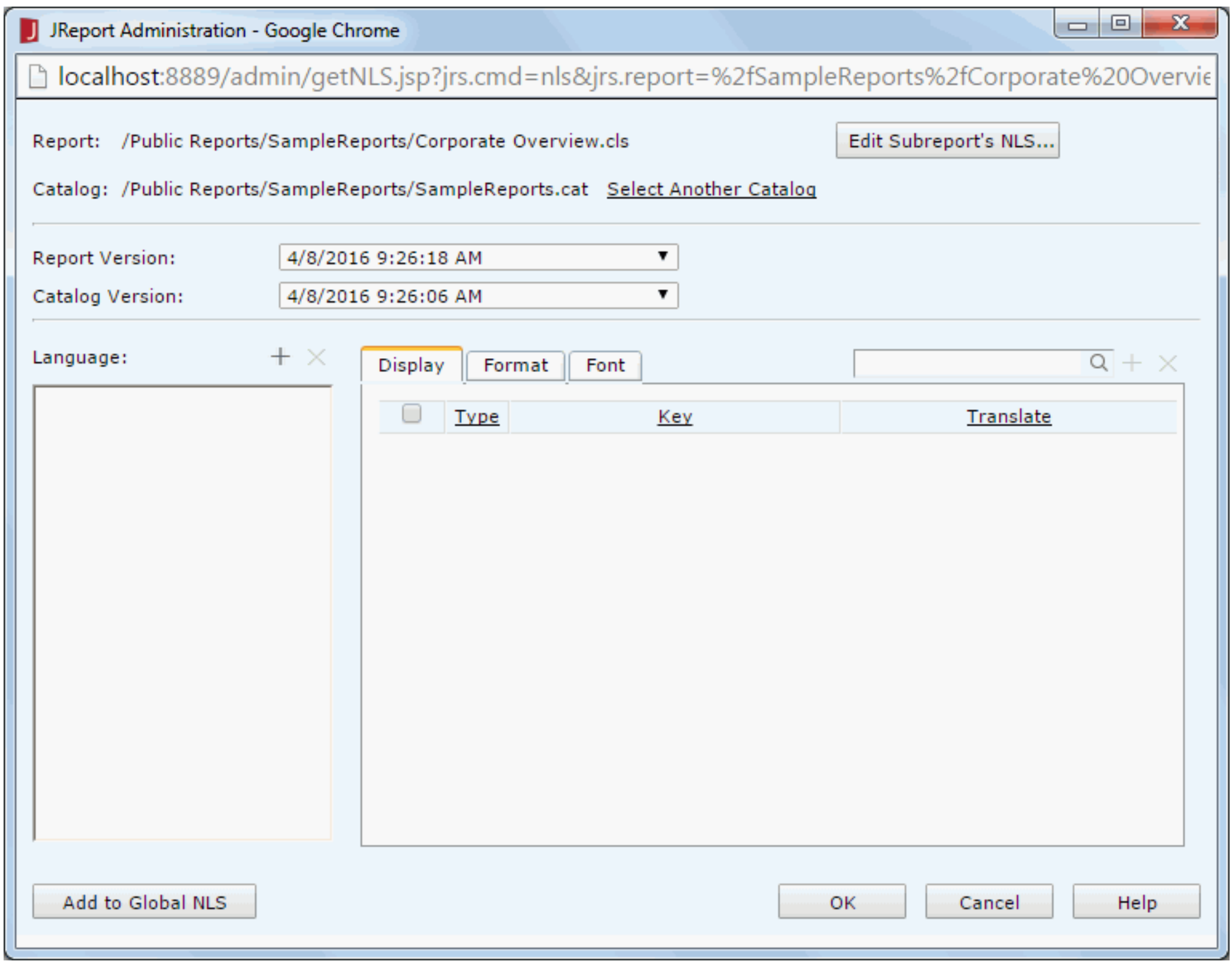
See also the Global NLS dialog for detailed explanation about options in the dialog.

Editing local NLS

JReport allows you to edit NLS for each catalog, report or library component as the report designer would with the NLS Editor in JReport Designer. With the NLS Editor, you can translate a catalog, report or library component into different languages from the original one. This feature is provided to administrators only.

To edit NLS for a catalog, report or library component on JReport Server:

1. Start JReport Server and log onto the JReport Administration page.
2. In the resource tree, browse to the catalog/report/library component for which you want to edit NLS.
3. Select the row that the catalog/report/library component is in, then click the **NLS Editor** button in the Control column. The NLS Editor appears.



4. Specify a report/component if you are editing NLS for a report/library component and catalog version as required.
5. Click the **Add** button above the Language box.
   - If you are editing NLS for a catalog, the Select Language Source dialog is displayed. Specify where to add the required languages, then click **OK**.
     - If Languages Supported by JReport is checked in the dialog, the Add Language dialog appears. Select the required languages and click **OK** to add them.
     - If NLS Resource File is checked in the dialog, the File Upload dialog appears. Browse to the local folder where the NLS resource file is located, select the file and then click **Open**.
   - If you are editing NLS for a report or library component, the Add Language dialog is displayed. Select the languages in which you want the report/library component to display, then click **OK** to confirm and go back to the NLS Editor.
6. The selected languages are now listed in Language box of the NLS Editor dialog. Select a target language from the box to edit NLS for it.
7. In the Display tab, click , then in the Add Display dialog, where all the display text in the catalog/report/library component are listed, select the keys you want to translate and enter the target language text for the keys in the Translate column, then click **OK**. The selected keys are now listed in the Display tab of the NLS Editor. You can further edit the translation for the keys here if required. Click **Add to Global NLS** to add the display information you just specify to the global NLS library of the target language if you want to reuse them later. If some of the keys already exist in the global NLS library, the Add to Global NLS dialog appears for you to handle the duplication.
8. Switch to the **Format** tab, click , then in the Add Format dialog, where all the formats used in objects of the catalog/report/library component are listed, select the keys you want to customize and provide the target language format for the keys in the Format column, then click **OK**. The selected keys are now listed in the Format tab of the NLS Editor. You can further edit the format for the keys here if required. Click **Add to Global NLS** to add the format information you just specify to the global NLS library of the target language if you want to reuse them later. If some of the keys already exist in the global NLS library, the Add to Global NLS dialog appears for you to handle the duplication.
9. Click the **Font** tab, click , then in the Add Font dialog, where all the fonts used in objects of the catalog/report/library component are listed, select the keys you want to customize and give the target language font face and font size for the keys in the Font Face and Font Size columns, then click **OK**. The selected keys are now listed in the Font tab of the NLS Editor. You can further edit the font face and font size for the keys here if required. Click **Add to Global NLS** to add the font information you just specify to the global NLS library of the target language if you want to reuse them later. If some of the keys already exist in the global NLS library, the Add to Global NLS dialog appears for you to handle the duplication.
10. Select another language and edit NLS for it as shown above.
11. Click **OK** to accept the settings.

Notes:

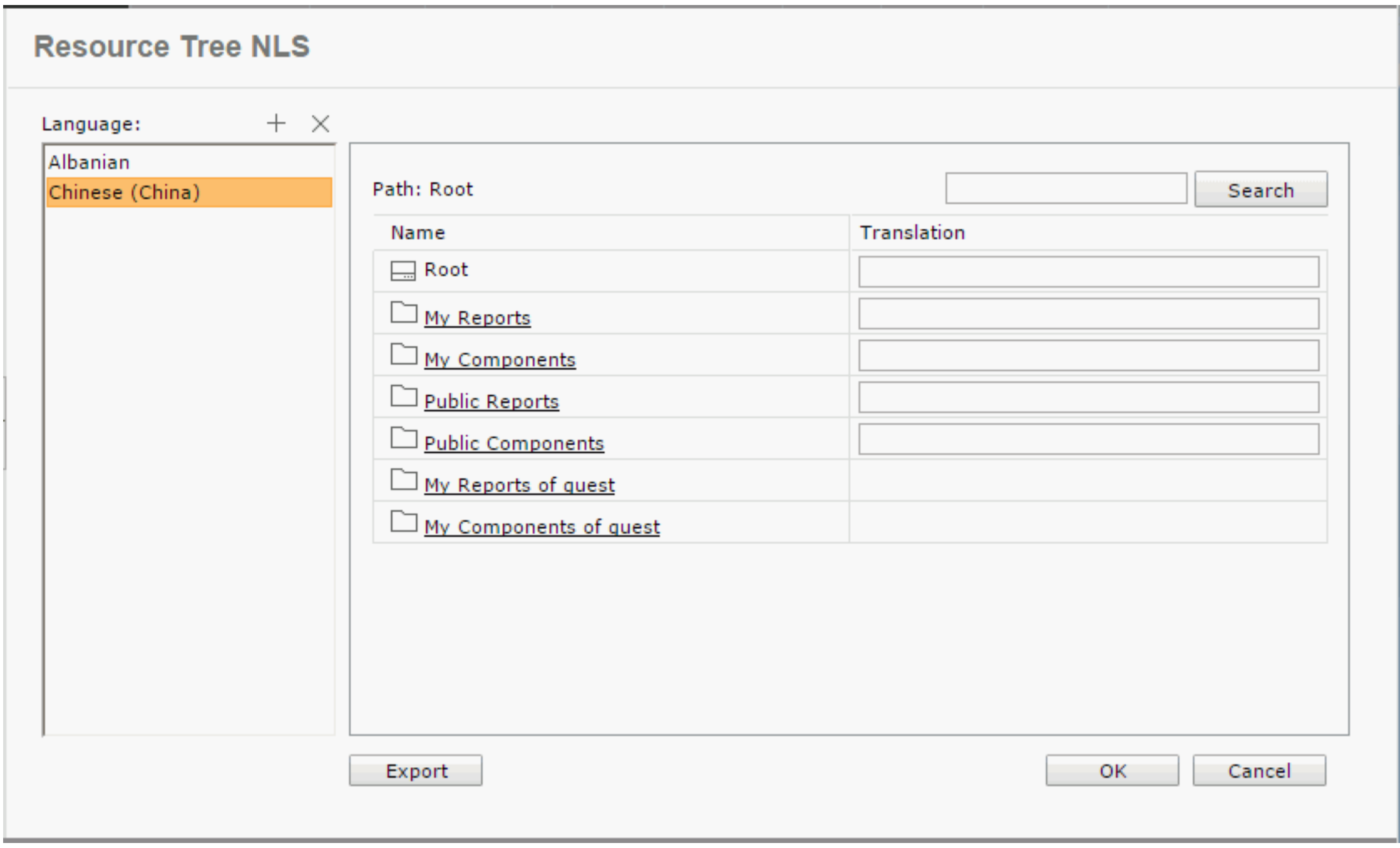
- The keys with default values will not be saved when you click the OK button in the NLS Editor.
- You can also edit NLS for a specific catalog/report/library component version. To do this, on the JReport Administration page, access the version table of the report/library component, then click the **NLS Editor** link for the catalog/report/library component version.
- When you switch among different languages by choosing languages from the Language box, you may find that the text in the Translate column become unreadable. To resolve this problem, you can add `-Djreport.url.encoding=UTF-8` to the batch file that starts the server and then restart it. This changes the encoding to Unicode which supports all languages.
- The Add to Global NLS option is not available to organization admin.

Editing resource tree NLS

JReport allows you to edit NLS for resources in the resource tree. With resource tree NLS, you can translate the names of all reports, library components and folders on the resource tree into different languages from the original one. This feature is provided to administrators only.

To edit NLS for resources in the resource tree on JReport Server:

1. Start JReport Server and log onto the JReport Administration page.
2. Click **Resources** on the system toolbar and select **Resource Tree NLS** from the drop-down menu. The Resource Tree NLS dialog appears.



3. Click the **Add** button + above the Language box. The Select Language Source dialog is then displayed.
4. Specify where to add the required languages.
   - **Languages Supported by JReport**
     Adds languages from the languages that JReport supports.
   - **NLS Resource File**
     Adds languages from an external NLS resource file which has been defined with some language information.

     NLS resource files should follow the naming rule: NLS_[language]_[region A2]_[User Defined].properties. The language argument is a valid ISO Language Code as defined by ISO-639. You can find a full list of these codes at a number of sites, for example: http://www.loc.gov/standards/iso639-2/php/code_list.php. The region argument is a valid ISO Country Code as defined by ISO-3166. You can find a full list of these codes at a number of sites, for example: http://www.chemie.fu-berlin.de/diverse/doc/ISO_3166.html.
5. Click **OK** in the Select Language Source dialog. Then,
   - If Languages Supported by JReport is checked in the dialog, the Add Language dialog will be displayed. Select the required languages and click **OK** to add them.
   - If NLS Resource File is checked in the dialog, in the displayed dialog, browse to the local folder where the NLS resource file is located, select the file and then click **Open**.

   The specified languages are now displayed in the Language box in the Resource Tree NLS dialog.
6. Select a language from the Language box to edit resource tree NLS for it.
7. The resource names are listed in the right panel. Give the corresponding target language text in the Translation column.
8. If needed, type the resource name you want to find in the Search box and then click the **Search** button, which helps you to find the resource quickly.
9. Click a folder to open resources in it. Give the corresponding target language text in the Translation column.

   When a new page is entered, the modifications you made on the last opened page will be saved automatically.
10. Repeat the above steps to define resource tree NLS for the other languages as required.
11. When done, click **OK** to accept the settings.
12. Click **Export** button to export the resource tree NLS map to a resource tree NLS file in the specified location for further use if needed.

    Note that only the resource tree of one translated language can be exported at a time and a user can only use the resource tree NLS file exported by himself.

Notes:

- The resource tree NLS takes effect only when the translated language is the same as the language specified for the option Specify Default Language in the Profile > Customize Server Preferences > Advanced dialog or the browser language, and the former has higher priority.
- Resource tree NLS is not supported for resources from a real path.

Running NLS reports/dashboards

When you have NLS edited for a report or library component on the JReport Administration page, or have an NLS report or library component published to the server from JReport Designer, you can then run the report or dashboard that contains the library component in the specified languages.

Directly running NLS reports/dashboards

To directly run an NLS report/dashboard in a specified language, follow the steps below:

1. On the JReport Console page, Click **Profile** on the system toolbar, then click **Customize Server Preferences** on the task bar of the Profile page.
2. Select the **Advanced** tab, check **Enable NLS** and choose the language from the Default Language drop-down list, in which you want the NLS report/dashboard to display by default, then select the corresponding encoding from the Default Encoding drop-down list.
3. Click **OK** to save the changes.
4. Click **Resources** on the system toolbar to switch to the Resources page.
5. Browse to the report/dashboard you want to run and click its name. The report result will then be displayed in the language you have specified.

Running NLS reports in Advanced mode or by scheduling

To run an NLS report in a specified language in Advanced mode:

1. On the JReport Console > Resources page, browse to the report you want to run, put the mouse pointer over the report row and click the **Advanced Run** button on the floating toolbar.
2. In the Format tab, select the **Enable NLS** checkbox, choose the language from the Using Language drop-down list, then select the corresponding encoding from the Encoding drop-down list.
3. Finish the other related options and click **Finish** to run the report. The report result will then be run in the selected language.

To schedule an NLS report to make it run in a specified language:

1. On the JReport Console > Resources page, browse to the report you want to schedule, put the mouse pointer over the report row and click the **Schedule** button on the floating toolbar.
2. In the General tab, select the **Enable NLS** checkbox, then choose the language from the Using Language drop-down list, select the corresponding encoding from the Encoding drop-down list.
3. Finish the other related options and click **Finish** to perform the task. The report result will then be run in the selected language.

Localizing page navigation links in HTML report outputs

When you schedule to publish a report to HTML format, or run it in Advanced mode in HTML format, you can localize the names of page navigation links in the HTML report outputs, such as First, Previous, Next, and Last, according to your requirements.

The localizing process is divided into three steps:

1. Create a property file for the desired language.
2. Enable the language for the report.
3. Apply the localized link names to HTML report outputs.

The following explain the steps in detail:

Creating the property file

To localize the page navigation link names in HTML report outputs, a property file must be created first for the desired language. To do this:

1. Create the sub directories in `<server_install_root>\resources` as follows: `<server_install_root>\resources\report\languages\[language-locale]\properties`. For example, `C:\JReport\Server\resources\report\languages\zh-cn\properties`.

   Click here for more information about the naming criterion for language package folders.

2. Create a report.properties file in the properties directory.
3. Open the property file and copy the following contents to it:

```
# The following is the report properties file format that can localize the link names in HTML.
4000101=First
4000102=Prev
4000103=Next
4000104=Last
4000105=Back
4000106=Refresh
4000107=@CurrentPageNumber; of @TotalPageNumber;
```

4. Translate the text after = to the language specified by the folder name.

   Note: For the line "`4000107=@CurrentPageNumber; of @TotalPageNumber;`", you just need to translate "of" to the desired language. In the HTML outputs, @CurrentPageNumber will be replaced by the current page number, and @TotalPageNumber by the report total page number.

5. Save the property file with UTF-8 encoding.
6. Copy the property file to the `<jdk_install_root>\bin` directory.

   Note: You can just add the `<jdk_install_root>\bin` directory to your PATH instead of copying the file.

7. Convert the contents in the property file into Unicode using native2ascii.exe in `<jdk_install_root>\bin` by running the following command:

```
C:\jdk1.6.0_17\bin>native2ascii -encoding utf-8 report.properties > newreport.properties
```

   Note: When you convert your property file to the same directory as the original one, you need to give it a new name instead of replacing the original in order to avoid problems.

8. Delete report.properties in `<server_install_root>\resources\report\languages\[language-locale]\properties` and copy newreport.properties in `<jdk_install_root>\bin` to it, then rename the property file back to report.properties.

Enabling the language for the report

When the property file is ready, the next step is to enable the language defined in the file for the required report.

1. Start JReport Server and log onto the JReport Administration page.
2. Browse to the row that the report is in, then click the **NLS Editor** button in the Control column.
3. In the NLS Editor dialog, specify a report and catalog version as required.
4. Click the button above the Language box, then in the Add Language dialog, choose the specified language and click **OK**.
5. Click **OK** in the NLS Editor dialog to confirm the settings.

Now the language will have been enabled for the report.

Applying the localized link names to HTML report outputs

To apply the localized link names to HTML outputs of the specified report, follow the steps below:

1. Log onto the JReport Console page and browse to the report.
2. Put the mouse pointer over the report row and click the **Advanced Run** button or **Schedule** button on the floating toolbar.
3. In the Format/General tab of the Advanced Run/Schedule dialog, check **Enable NLS** and select the specified language from the Using Language drop-down list.
4. Specify the other settings and finish the task. Then in the generated HTML outputs, you can see that the page navigation links are displayed in the language you defined for the property file.

Tuning Performance

Many variables affect the performance of JReport Server. For example, the hardware and software environment that it runs in. You may or may not have any influence in this regard, but you are able to carry out performance tuning to make JReport Server efficient, reliable, and fast.

The following lists the factors that help with the server performance:

- Use Connection Pooling either by using JNDI and the connection pool of the application server or JReport connection pooling using ConnectionPoolConfig.properties. See Configuring connection pool.
- For best performance and reliability you should change the JReport Server DBMS from Derby (the internal Java DBMS) to your standard production DBMS or other DBMS you are familiar with. Derby is included for out of the box ease of use but is not recommended for production use. See Configuring the server database on the JReport Administration page.
- Limiting the number of actively running reports can avoid momentary system overload. Setting the property performance.max.reports to a limited number in server.properties in the `<install_root>\bin` directory can smooth out performance and provide higher overall throughput. The best number depends entirely on your environment so needs to be configured after testing which setting provides the highest throughput in your environment.
- When using an application server, you can achieve a similar result to performance.max.reports by limiting the total number of active threads for JReport. You can assume a report consumes around 5 threads on average so setting the number of active threads to 50 is similar to setting performance.max.reports=10 but may provide better overall throughput since it is has finer granularity to control CPU utilization.
- Cache catalogs, reports, and images by configuring the Cache panel on the JReport Administration page. This is effective when the same catalogs, templates and images are used frequently but may harm performance when many different catalogs, templates and images are used and the requested resource is not in the cache. In this case, it slows performance checking the cache and wastes memory.
- Preload catalogs and reports by configuring the Configuration > Performance panel on the JReport Administration page. This will improve the performance the first time the report or catalog is used.
- Set log trace level and error level to record less information. In production servers trace level should be OFF and error level should be ERROR unless you are searching for specific issues such as monitoring the SQL statements sent to the DBMS.
- Convert reports from earlier versions into current version.
- Set task-level timeout for advanced run and scheduled tasks. This will ensure that users do not run extremely long jobs accidentally.
- Modify the JVM heap size. Again this needs some testing on your system to determine the best overall throughput. On 32-bit systems the maximum size is 2GB; however, on 64-bit systems the maximum size is extremely large. The heap usage can be monitored by tools such as the JDK jconsole application. Set a starting heap size just under your normal average usage (Use -Xms in JRServer.sh/bat) and set the maximum heap (Use -Xmx) to the maximum that you want the heap to grow. JReport can use additional heap to improve performance of queries and sorting and reducing the number of garbage collections (GC) which use CPU time. However, using too much memory for Java heap can impact performance of other applications.
- Set queue.policy=1, 2 or 3 in server.properties in the `<install_root>\bin` directory to turn on using priorities so on demand reports will be higher than scheduled reports. By using the different policies, you can dynamically adjust priorities so reports increase their priority the longer they wait in the queue. A third high priority queue can also be enabled for on demand reports which need to run immediately.
- Configure dhtml.properties to queue page report actions to ensure Page Report Studio features do not take too many system resources and impact other functions.
- Set engine.single_thread=true or false in server.properties. Depending on your environment and JDK, some systems run faster with a single threaded engine than with a multi-threaded engine. This needs to be tested in your environment to determine which setting provides the highest throughput.
- Set the maximum number of user handlers to the optimal size by setting the httpserver.max.handlers property in the server.properties file. This is the number of threads which are waiting to handle user requests. Setting this larger than needed wastes resources; however, setting it too small will mean that some users will not be able to access reporting until someone else exits.
- Adjust the engine's utilization of the CPU in accordance with your own requirements, by setting the Engine Priority property to a higher or lower priority on the JReport Administration page > Configuration > Advanced panel.
- Select the two options View Incomplete Pages and Format Page on Demand on the JReport Administration/Console page > Profile > Customize Profile > Page Report Studio > Properties > Advanced tab for faster viewing of the first page of large reports in Page Report Studio. The disadvantage is that to see the actual number of pages, calculate complete report totals and create the TOC, the user has to manually select the last page of the report.
- Be sure performance.exe_gc=false as the default in the server.properties file. This allows the JDK to schedule garbage collections as needed rather than at specific times.
- Minimize the usage of Word Wrap to columns which really require it. Word Wrap requires that we pre-format the data in memory to determine its size then wrap the field as necessary then continue to format it. If fields never are large enough to wrap, especially objects like labels, it is a lot of overhead.
- True Type Fonts (TTF) provide better throughput by reducing font substitution and scaling time. Changing all the fonts to use TTF fonts improves the quality of the output as well as the performance when a mixture of operating system is used such as report design on Windows and production on Unix.
- Use binary versions of .cls and .cat, not the xml versions when you publish reports to JReport Server. This saves processing time and I/O time when converting the files to Java classes.
- Customize heartbeat interval for web browser products such as Web Report Studio, Page Report Studio, JDashboard, and Visual Analysis.

  The heartbeat mechanism is introduced in Web Report Studio, Page Report Studio, JDashboard, and Visual Analysis. It works together with user session timeout to release the unused resources of the products in time. It begins to work after any of the products is opened, by reporting to JReport Server that the product is alive per a period of time in a way similar to heartbeats. If within 4 times the heartbeat interval JReport Server has not received a heartbeat, the product will be expired and the resources be released. Heartbeat stops in the following situations:

  - The product or the related web browser is closed.
  - The computer is powered off.
  - Though the product is open, there are not any actions happening to the product such as mouse move and automatic refresh until the session times out.

  If the heartbeat interval is shorter than the user session timeout time which can be configured by the administrator on the JReport Server Administration page > Configuration > Advanced panel, the session will not expire as long as there are heartbeats. If the heartbeat interval is longer than the session timeout time, the session will be expired when the session times out.

  By default heartbeat is activated and the heartbeat interval is 15000 milliseconds. The heartbeat can be configured by adding the JVM parameter -Djrd.heartbeat in the JReport Server startup file. The parameter is used to specify the heartbeat interval in milliseconds. For example, -Djrd.heartbeat=95000.

  Heartbeat use Web Worker technology of HTML5. Web Worker is supported by Firefox, Chrome, Safari and IE10 where it is working in a separate javascript thread. But in IE9, IE8 JReport can only simulate Web Worker using iFrame, which means that the heartbeat code can be blocked by render code in IE 9 browsers.

Monitoring JReport Server

JReport Server Monitor, which is a standalone web-based application used to monitor the overall performance of JReport Server, allows you to inspect the status of JReport Server, including the status of the servers in the cluster, the status of different reports, the status of on-line users and so on. JReport Server Monitor is able to generate and display the performance chart of JReport Server according to its statistics. Thus you can view the performance of JReport Server in the form of Line Chart Graph and Text. You can also use JReport Server Monitor to maintain JReport Server, such as shutting down servers, stopping waiting/running tasks, logging out a valid user session and so on. By creating profiling reports using JReport Server Monitor, you can inspect server performance in a certain period of time.

To access JReport Server Monitor:

1. Launch MonitorServer.bat in `<monitor_install_root>\bin`.
2. Click **Monitor** on the system toolbar of the JReport Administration page.

The following topics detail how to use the Server Monitor:

- Monitoring the server status
- Monitoring the server performance

Monitoring the server status

JReport Server Monitor allows you to inspect the status of JReport Server. You can list the servers and their status from a cluster. You can also list the reports by drilling down into the servers, for example, running reports, waiting reports, finished reports, and failed reports. Tracking further down, you can even view the status of these reports. However, JReport Server Monitor is not able to monitor the information of a report that is running in Page Report Studio.

The following topics explain what JReport Server Monitor can do:

- Showing the status of servers in a cluster
- Monitoring the status of reports
- Monitoring the status of online users
- Monitoring the status of the database connection pool

Showing the status of servers in a cluster

By accessing the home page of JReport Server Monitor, you can see the status of each server in a cluster, including the cluster member ID, host IP, port, and its status.

The Status table of the servers includes:

Column Heading	Description
Cluster Member ID	The ID of the server as a cluster member.
Host	The host IP address of the server.
Port	The RMI port number of the server.
Status	The status of the server. Can be either Active or Inactive. • **Active** - The server has been started and is ready for service. • **Inactive** - The server is inactive and cannot be available for service.

Monitoring the status of reports

Expand any server node in the left panel of the JReport Server Monitor home page, and then click **Reports**, you can see the status of the page reports, web reports and dashboards on the server.

There are five types of report status: all reports, running reports, waiting reports, finished reports, and failed reports. You can select to view the status of different reports from the drop-down list.

Status of all reports

The status table of all reports includes:

Column Heading	Description
Report	The full path name of the report.
User ID	The ID of the user who opened the report.
Submit Time	The time when the report was last opened.
Pages	The total number of pages the last opened report has. Not available for web reports and dashboards.
Number of Runs	The total number of report runs since it is first published to JReport Server.
Status	The status of the opened report.

In addition, when you click the full path name of a report, the Task Statistics dialog will pop up, showing you the detailed task statistics, which includes the following:

Task ID

The exact date and time when the report was opened.

Report

The path and name of the opened report.

Catalog

The catalog where the opened report lists.

Total Number of Times Run

The total number of times the opened report has run ever since a specific time.

Average Number of Times per Day

The average number of times the opened report has run per day.

Last Submit Time

The time when the report was last opened.

Status of the running reports

The status table of the running reports includes:

Column Heading	Description
Action	Stops the report from running and makes it a failed report.
Task ID	The internal ID for this report.
Report	The full path name of the report.
User ID	The ID of the user who opened the report.
Task Status	For running reports, can be one of the following: • **Running** - The task is currently being processed. • **Initializing Engine** - The task is currently in the initializing engine stage. • **Loading Report** - The task is currently in the loading report stage. • **Exporting** - The task is currently in the exporting stage. • **Exiting Engine** - The task is currently in the exiting engine stage.
Task Type	The task type. Can be one of the following: Schedule, On-Demand, Page Report Studio, Web Report Studio, Dashboard and Bursting.
Start Time	The time when the task was started.
Elapsed Time	The elapsed time since the task was started.
Submit Time	The time when the report was opened.
Run Host	The name of the host on which the task is performed.
Run Port	The port number of the host on which the task is performed.
Catalog	The catalog that the report belongs to.

In addition, when you click the task ID of a report, the Task Information dialog will pop up, showing you the detailed task information, which includes the following:

Task ID

The exact date and time when the report was opened.

Task Type

The task type. There are six task types: Schedule, On-Demand, Page Report Studio, Web Report Studio, Dashboard, and Bursting.

Report

The path and name of the opened report.

Catalog

The catalog where the opened report lists.

Report Pages

The total page number of the report. Not available when task type is Web Report Studio or Dashboard.

Submit Time

The time when the report was opened.

Start Run Time

The time when the report was started to run.

Completed Time

The time when the report was closed.

Parameters

The parameters that user uses to run the report. If the report runs with no parameters, this column will be left empty.

Status of the waiting reports

The status table of the waiting reports includes:

Column Heading	Description
Action	Stops the report from running and makes it a failed report.
Task ID	The internal ID for this report. If you click the ID, the Task Information dialog will pop up showing you the detailed task information.
Report	The full path name of the report.
User ID	The ID of the user who opened the report.
Task Status	For waiting reports, can be one of the following: • **Submitted** - The task has been submitted successfully. • **Unlaunch** - The task is currently in the unlaunch queue waiting to be processed. • **Task Queue** - The task is currently in the task thread queue waiting to be processed.
Task Type	The task type. Can be one of the following: Schedule, On-Demand, Page Report Studio, Web Report Studio, Dashboard, and Bursting.
Submit Time	The time when the report was opened.
Catalog	The catalog that the report belongs to.

Status of the finished reports

The status table of the finished reports includes:

Column Heading	Description
Task ID	The internal ID for this report. If you click the ID, the Task Information dialog will pop up showing you the detailed task information.
Report	The full path name of the report.
User ID	The ID of the user who opened the report.
Task Status	Completed. The task has been processed successfully, and has accomplished all the requirements.
Task Type	The task type. Can be one of the following: Schedule, On-Demand, Page Report Studio, Web Report Studio, Dashboard and Bursting.
Run Host	The name of the host on which the task is performed.
Run Port	The port number of the host on which the task is performed.
Catalog	The catalog that the report belongs to.
Result Files	The path of the result files.
Report Pages	The total page number of the report. Not available when task type is Web Report Studio or Dashboard.
Reason	The reason why the task failed. Can be an exception or a meaningful description.
Submit Time	The time when the report was opened.
Start Run Time	The time when the report was started to run.
Completed Time	The time when the report was closed.

Status of the failed reports

The status table of the failed reports includes:

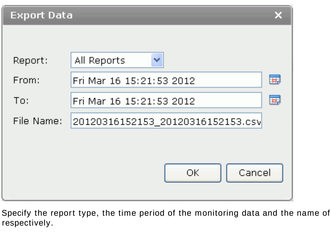
Column Heading	Description
Task ID	The internal ID for this report. If you click the ID, the Task Information dialog will pop up showing you the detailed task information.
Report	The full path name of the report.
User ID	The ID of the user who opened the report.
Task Status	Failed. The task has encountered errors, and has failed to accomplish all the requirements.
Task Type	The task type. Can be one of the following: Schedule, On-Demand, Page Report Studio, Web Report Studio, Dashboard, and Bursting.
Run Host	The name of the host on which the task is performed.
Run Port	The port number of the host on which the task is performed.
Catalog	The catalog that the report belongs to.
Result Files	The path of the result files.
Report Pages	The total page number of the report. Not available when task type is Web Report Studio or Dashboard.
Reason	The reason why the task fails. Can be an exception or a meaningful description.
Failed Info	The information about the report's failure.
Submit Time	The time when the report was opened.
Start Run Time	The time when the report was started to run.
Completed Time	The time when the report was closed.

Exporting the monitoring data

You can export the monitoring data of the reports' status to a CSV file. However, before doing this, you need to make sure the profiling DB on the server node where the monitoring data will be saved has already been configured. For details about how to configure the profiling DB, see the topic Configuring the server database in a standalone environment.

To export the monitoring data of the reports' status:

1. Click the **Export Data** link on the reports status panel. The Export Data dialog appears.



2. Specify the report type, the time period of the monitoring data and the name of the exported file respectively.
3. Click **OK** to export the data.

The following are details about options in the Export Data dialog:

Report

The type of reports you want to export the monitoring data.

From

The date and time from which the monitoring data starts. You can type it in the text field or click the calendar button to select the date and time.

To

The date and time to which the monitoring data ends. You can type it in the text field or click the calendar button to select the date and time.

File Name

The name of the exported file.

OK

Exports the data to a CSV file and opens the File Download panel. You can choose to open the file or save it elsewhere.

Cancel

Does not retain any changes and closes the dialog.

Note: The exported monitoring data in the CSV file are encoded with UTF-8. If you open an exported CSV file which contains Chinese strings by a double-click, the Chinese strings would be displayed as random code in the file. To solve this problem, you need to open the file in the following way:

1. Start Microsoft Excel first.
2. Click **From Text** in the Data tab, then select the exported CSV file and click **Import**.
3. Select **65001: Unicode (UTF-8)** in the File Origin selector, then keep clicking **Next** and at last click **Finish** and **OK**.

Monitoring the status of online users

To see the status of the online users, expand any server node in the left panel of the JReport Server Monitor home page, and then click **Users**. You can also select and remove specific users.

The status table of the online users includes:

Column Heading	Description
Session ID	The internal ID of the user session.
User ID	The ID of the user logged onto the server.
Create Time	The time when the user session was created.
Last Access Time	The time when the user last accessed the server.
HTTP Session ID	The session ID in the HTTP service.
Authentication	The authentication type. It can be Internal or External.

Monitoring the status of the database connection pool

To see the status of the database connection pool, expand any server node in the left panel of the JReport Server Monitor home page, and then click **Databases**. You can also select and remove specific connections.

The status table of the connection pool includes:

#	Column Heading	Value	Description
1	User	String	The user who is currently using the connection.
2	URL	A URL connecting to a database.	It specifies the connections that are based on a URL which will be caught in the pool.
3	Expiring Time(s)	0 (default) or expiring time. The unit is second. If the value is zero then the connection will never expire.	Shows the time during which a connection can be alive. If a connection has expired, the connection pool will close it.
4	Idle Expiring Time (s)	1 (default) or expiring time. The unit is second. Its value must be larger than or equals to 1.	Shows the time during which a connection is kept after it starts idling. If a connection is not used, it will stay open until the idle expiry time has been reached.
5	Maximal Connection Count	0 (default) or a positive integer.	Shows the pool size, which limits the number of connections under a single URL
6	Maximal Share Count	0 (default) or a positive integer.	Shows the number of users who can share a connection simultaneously.
7	Attempt	A positive integer, the default value is 1.	Shows whether to re-create a connection when the connection pool has failed to create one and the number of attempts for creating the connection. If the user sets this value to a non-positive integer, the default value (1) will be used.
8	Interval(ms)	0 (default) or a positive integer. The unit is millisecond (ms).	If property 'Attempt' is larger than 1, then before the connection pool retries to create a connection it will wait for an interval time. This property defines the interval time.
9	Last User Connecting Time(s)	0 (default) or a positive integer. The unit is second.	It shows the time elapsed since the last user has taken this connection.
10	Current Idle Time (s)	0 (default) or a positive integer. The unit is second.	Shows the time elapsed since the connection started to idle.
11	Current Share Count	0 (default) or a positive integer.	Shows the number of users who are currently sharing this connection.

Note: The properties numbered 2 to 8 can be set in the ConnectionPoolConfig.properties file in `<server_install_root>\bin` and the last three properties will be shown according to the real time status.

Monitoring the server performance

You can monitor the server performance in two ways, one is using JReport Server Monitor, and the other is by using an external method: JMX MBean API or web service API.

Monitoring the server performance on JReport Server Monitor

JReport Server Monitor can show performance counters in graph (Line chart and Bar chart) and text mode.

To monitor the performance of JReport Server on JReport Server Monitor:

1. In the left panel of the JReport Server Monitor home page, click to select any server node.
2. Click the **Performance** tab and the performance chart of the specified server will be displayed.
3. If you want to configure the performance chart, click the **Graph Options** button on the toolbar. In the Graph Options dialog, set how many tick marks you need to display on the X axis in the Keep Last N Records text field. In the Y Axis Limit and Y2 Axis Limit text fields, set respectively the maximum value on the left Y axis and the right Y axis. In the Interval text field, set the time interval (in seconds) the performance chart will use to get data and refresh itself.

   Click the **Clear Display** button if you want to clear the current display of the performance counters in the performance chart.

   To stop the current display of the performance counters, click the **Stop** button on the toolbar.

The following are the available counters:

Performance Counter	Description
Waiting Reports	The number of the currently waiting reports.
Running Reports	The number of the currently running reports.
Finished Reports	The number of the finished reports.
Finished Report Pages	The number of pages of the finished reports.
Report Average Processing Time	The average processing time of each report.
Report Average Waiting Time	The average waiting time of each report.
Valid User Sessions	The number of valid user sessions.
Average Submitted Tasks per User	The average number of tasks that each user has submitted since JReport Server started.
Database Connections	The number of database connections.

Monitoring the server performance using JMX MBean API or web service API

To monitor server performance via JMX MBean API:

1. Make your application environment consistent with the JMX specification.
2. In server.properties, set the properties server.rmiserver.enable and server.profiling.enable to true.
3. Run your code. Here is an example:
```
ObjectName profilingName = new ObjectName("jinfonet:name=profileCounter");
ProfilingFactory ft = new ProfilingFactory("127.0.0.1",1129, "d:\\monitor");
ProfilingCounter profilingCounter = new ProfilingCounter(ft);
MBeanServer server = MBeanServerFactory.createMBeanServer();
server.registerMBean(profilingCounter, profilingName);
ProfileNotifyListener notifyListener = new ProfileNotifyListener();
profilingCounter.addNotificationListener(notifyListener, null,null);
// Create an RMI connector and start it
JMXServiceURL url = new JMXServiceURL("service:jmx:rmi:///jndi/rmi://localhost:9999/server");
JMXConnectorServer cs = JMXConnectorServerFactory.newJMXConnectorServer(url, null, server);
cs.start();
```

   In code part `ProfilingFactory ft = new ProfilingFactory("127.0.0.1",1129, "d:\\monitor")`, the parameters serverHost and serverRMIPort are used to get server's ProfilingService remote object, parameter homepath is the path of the rmi.auth file which is used for RMI authority checking public ProfilingFactory(String serverHost, int serverRMIPort, String homepath).
4. Check the monitoring result.

To monitor server performance via web service API:

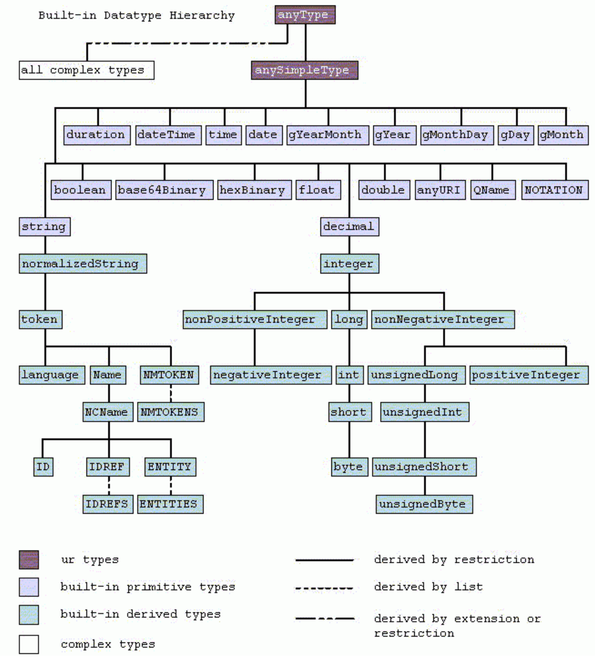
1. Build jreport.war with the `makewar buildWar4WS` command by using the makewar.bat utility in the `<install_root>\bin` directory.
2. Deploy jreport.war to your application server such as Tomcat or WebSphere.
3. In server.properties, set the properties server.rmiserver.enable and server.profiling.enable to true.
4. Run your code. Here is an example:
```
XFire xfire = XFireFactory.newInstance().getXFire();
XFireProxyFactory factory = new XFireProxyFactory(xfire);
Class className = jet.server.profiling.ws.service.JReportServerProfilingWs.class;
String serviceName = "ProfilingService";
String serviceURL = "http://localhost:8080/jreport/services/ProfilingService";
Service serviceModel = new ObjectServiceFactory().create(className, serviceName, null, null);
JReportServerProfilingWs o = (JReportServerProfilingWs)factory.create(serviceModel, serviceURL);

//Setup properties
Client client = Client.getInstance(o);
client.setProperty("mtom-enabled", "true");
client.setProperty(HttpTransport.CHUNKING_ENABLED, "true");
//try call getCounter first before check login
try{
   System.getCounter("totalcompletedtasks"));
}catch(Exception e) {
   e.printStackTrace();
}
//call check login then call getCounter and getCounterType
o.checkLogin("admin", "admin");
System.getCounter("totalcompletedtasks"));
```
5. Check the monitoring result. When you run your code, you can call the two methods getCounter() and getCounterType() to get the information you want.

The available counters used in JMX MBean API and web service API are:

Performance Counter	Counter Name	Counter Type	Description
Total Completed Tasks	TotalCompletedTasks	int	The tasks that are totally completed.
Number of Reports per Minute	NumberofReportsPerMinute	float	The number of the reports running per minute.
Total Number of Pages Exported	TotalNumberofPageExported	long	The total number of the exported report pages.
Exported Pages per Minute	ExportPagesPerMinute	float	The number of exported report pages per minute.
Successful Tasks	SuccessfulTasks	int	The tasks that have been run successfully.
On-demand Tasks	OnDemandTasks	int	The tasks that have been run on-demand.
Average Number of Pages	TaskAveragePages	float	The average number of report pages for each task.
Maximum Number of Pages	TaskMaximumPages	long	The maximum number of report pages for all tasks.
Average Running Time per Task	TaskAverageRunTime	float	The average running time of each task.
Maximum Running Time	TaskMaximumRunTime	float	The maximum running time of the tasks.
Average Waiting Time per Task	TaskAverageWaitTime	float	The average waiting time of each task.
Maximum Waiting Time	TaskMaxWaitTime	float	The maximum waiting time of the tasks.
Average Engine Time per Task	TaskAverageEngineTime	float	The average running time of each task on engine.
Maximum Engine Time	TimeTaskMaximumEngine	float	The maximum running time of all tasks on engine.
Average Concurrent Engines	AverageConcurrentEngines	float	The average number of concurrent engines for each task.
Maximum Concurrent	MaximumConcurrent	long	The maximum number of concurrent engines for all tasks.

In JMX MBean API and web service API, the data type of the profiling counters has to be Decimal. Below is a figure of built-in data type hierarchy that you could refer for the return value:



Working with JReport Server via URL

JReport Server provides URL commands for running and working with report resources. These URL commands are normally not typed by the user into the browser but are provided by your application which calls JReport.

- Commands for working with server resources via URL
- Authentication parameters
- Accessing the report creation wizard via URL
- Running reports via URL
- Scheduling reports via URL
- Specifying parameter values using dialog when running or scheduling reports via URL

Commands for working with server resources via URL

With the commands below, you can directly type the whole URL in a web browser or create it in your web application to access the destination. The HTTP methods GET and POST are available for almost all of the commands.

- **jrs.get_cat_rpts_new|jrs.get_subnodes**
  Description: Gets all resource nodes (folder, catalog, report and result) of a folder.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.get_cat_rpts_new&jrs.path=RESOURCENODE
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.get_cat_rpts_new&jrs.path=/SampleReports
  Response: Resource nodes list.
- **jrs.get_node_prop**
  Description: Gets the properties of a resource node.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.get_node_prop&jrs.path=RESOURCENODE
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.get_node_prop&jrs.path=/SampleReports/SampleReports.cat

  Response: Resource node properties.
- **jrs.delete_resource**
  Description: Deletes the resource node from the resource tree.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.delete_resource&jrs.path=RESOURCENODE
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.delete_resource&jrs.path=/SampleReports/Employee Information.cls
- **jrs.get_rst_vers**
  Description: Gets a result version list of a report.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.get_rst_vers&jrs.catalog=CATALOG&jrs.report=REPORT_SET
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.get_rst_vers&jrs.catalog=/SampleReports/SampleReports.cat&jrs.report=/SampleReports/Employee Information.cls
  Response: Result versions list.
- **jrs.get_rpt_vers**
  Description: Gets a report version list of a report.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.get_rpt_vers&jrs.report=REPORT
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.get_rpt_vers&jrs.report=/SampleReports/Employee Information.cls
  Response: Report version list.
- **jrs.get_cat_vers**
  Description: Gets a catalog versions list of a catalog.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.get_cat_vers&jrs.catalog=CATALOG
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.get_cat_vers&jrs.catalog=/SampleReports/SampleReports.cat
  Response: Catalog versions list.
- **jrs.get_rst_doc_vers**
  Description: Gets a result versions list of a result document. The result document can be generated by advanced running or scheduling a report.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.get_rst_doc_vers&jrs.result=RESULT
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.get_rst_doc_vers&jrs.result=/SampleReports/Employee Information
  Response: Result versions list.
- **jrs.get_ver_rst_page**
  Description: Gets a result version of the report from the version manager.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.get_ver_rst_page&jrs.file=FILE_NAME
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.get_ver_rst_page&jrs.file=1%5cJReport_System_User32740635%5cInvoiceReport.rst
  Response: The result file from the version manager.
- **jrs.get_ver_rst**
  Description: Gets an HTML page for viewing a result version of a report with the ViewerApplet.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver/REPORT?jrs.cmd=jrs.get_ver_rst&jrs.file=FILE_NAME
  URL Example: http://localhost:8888/jrserver/SampleReports/SampleReports.cat/InvoiceReport.cls?jrs.cmd=jrs.get_ver_rst&jrs.file=1%5cJReport_System_User32740635%5cInvoiceReport.rst
  Response: HTML page for viewing the result file with the ViewerApplet.
- **jrs.get_ondemands**
  Description: Gets a list of reports running in background.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.get_ondemands
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.get_ondemands
  Response: Background run report list.
- **jrs.view_ver_rst**
  Description: Views a result version of a report.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver/REPORT?jrs.cmd=jrs.view_ver_rst&jrs.hist_file=RESULT_VERSION_FILE_NAME&jrs.result_type=RESULT_TYPE&jrs.applet_type=APPLET_TYPE[&jrs.isjrsapltpp=TRUE/FALSE][&jrs.is_multi_files=TRUE/FALSE][&jrs.is_applet_chart=TRUE/FALSE][&jrs.has_hyperlink=TRUE/FALSE][&jrs.has_page_number=TRUE/FALSE][&jrs.use_table=TRUE/FALSE][&jrs.drilldown=TRUE/FALSE][&jrs.relative_font_size=TRUE/FALSE][&jrs.no_margin_html=TRUE/FALSE][&jrs.web_browser=BROWSER_TYPE][&jrs.no_margin_pdf=TRUE/FALSE][&jrs.is_norm_txt=TRUE/FALSE][&jrs.delimiter=DELIMITER][&jrs.udchar_width=NUMBER][&jrs.udchar_height=NUMBER][&jrs.no_margin_ps=TRUE/FALSE][&jrs.no_margin_rtf=TRUE/FALSE]

  URL Example: http://localhost:8888/jrserver/SampleReports/SampleReports.cat/Employee Information.cls?jrs.cmd=jrs.view_ver_rst&jrs.hist_file=1%5cJReport_System_User32740635%5cEmployee Information.rst&jrs.result_type=1

  Notes:

  - RESULT_TYPE:
    0 - To Applet
    1 - To HTML
    2 - To PDF
    3 - To TEXT
    4 - To Excel
    5 - To PostScript
    6 - To Rich Text Format
    7 - To XML
    8 - To Page Report Result
  - APPLET_TYPE:
    2 - Java Plug-In 1.2 for Windows
    3 - Java Plug-In 1.3 for Windows
  - BROWSER_TYPE:
    0 - IE or Chrome
    1 - Firefox
- **jrs.get_rpt_desc_page**
  Description: Gets an HTML page for entering parameter values and choosing the result type of a report.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.get_rpt_desc_page&jrs.catalog=CATALOG&jrs.report=REPORT
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.get_rpt_desc_page&jrs.catalog=/SampleReports/SampleReports.cat&jrs.report=/SampleReports/InvoiceReport.cls
- **jrs.get_rpt_param_page**
  Description: Gets an HTML page for entering parameter values of a report.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.get_rpt_param_page&jrs.catalog=CATALOG&jrs.report=REPORT
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.get_rpt_param_page&jrs.catalog=/SampleReports/SampleReports.cat&jrs.report=/SampleReports/InvoiceReport.cls
- **jrs.del_rpt_ver**
  Description: Deletes a version of a report.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver/REPORT?jrs.cmd=jrs.del_rpt_ver&jrs.version_id=VERSION_ID
  URL Example: http://localhost:8888/jrserver/SampleReports/SampleReports.cat/InvoiceReport.cls?jrs.cmd=jrs.del_rpt_ver&jrs.version_id=103
- **jrs.del_rst_ver**
  Description: Deletes a result version of a report.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver/REPORT?jrs.cmd=jrs.del_rst_ver&jrs.version_id=VERSION_ID
  URL Example: http://localhost:8888/jrserver/SampleReports/SampleReports.cat/InvoiceReport.cls?jrs.cmd=jrs.del_rst_ver&jrs.version_id=106
- **jrs.del_rstdoc_ver**
  Description: Deletes a version of a result document.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.result=RESULT&jrs.cmd=jrs.del_rstdoc_ver&jrs.version_id=VERSION_ID
  URL Example: http://localhost:8888/jrserver?jrs.result=/SampleReports/InvoiceReport&jrs.cmd=jrs.del_rstdoc_ver&jrs.version_id=108
- **jrs.get_ver_param**
  Description: Gets the parameter file of a report's result version.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver/REPORT?jrs.cmd=jrs.get_ver_param&jrs.version_id=VERSION_ID
  URL Example: http://localhost:8888/jrserver/SampleReports/SampleReports.cat/InvoiceReport.cls?jrs.cmd=jrs.get_ver_param&jrs.version_id=103
- **jrs.get_schedules**
  Description: Gets the schedules list.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.get_schedules
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.get_schedules
- **jrs.del_schedule**
  Description: Deletes a schedule.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.del_schedule&jrs.task_id=TASK_ID
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.del_schedule&jrs.task_id=2003-12-11 11:09:16.455
- **jrs.enable_schedule**
  Description: Enables a schedule.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.enable_schedule&jrs.task_id=TASK_ID
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.enable_schedule&jrs.task_id=2003-12-11 11:09:16.455
- **jrs.disable_schedule**
  Description: Disables a schedule.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.disable_schedule&jrs.task_id=TASK_ID
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.disable_schedule&jrs.task_id=2003-12-11 11:09:16.455
- **jrs.get_new_schd_page**
  Description: Gets an HTML page in order to create a new schedule for the report with a web browser.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.get_new_schd_page&jrs.catalog=CATALOG&jrs.report=REPORT
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.get_new_schd_page&jrs.catalog=/SampleReports/SampleReports.cat&jrs.report=/SampleReports/Employee Information.cls
- **jrs.get_edit_schd_page**
  Description: Gets an HTML page to edit the schedule with a web browser.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.get_edit_schd_page&jrs.task_id=TASK_ID
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.get_edit_schd_page&jrs.task_id=2003-12-11 11:13:23.02
- **jrs.get_completed**
  Description: Gets the completed tasks list.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.get_completed
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.get_completed
- **jrs.del_completed**
  Description: Deletes a completed task.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.del_completed&jrs.id=RECORD_ID
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.del_completed&jrs.id=1071119097006-154848108
- **jrs.del_all_completed**
  Description: Deletes all completed tasks.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.del_all_completed
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.del_all_completed
- **jrs.get_active**
  Description: Gets the active tasks list.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.get_active
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.get_active
- **jrs.stop_task**
  Description: Stops an active task.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.stop_task&jrs.task_id=TASK_ID
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.stop_task&jrs.task_id= 2003-12-11 11:13:23.02
- **jrs.get_change_password_page**
  Description: Gets an HTML page in order to change the password with a web browser.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.get_change_password_page
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.get_change_password_page
- **jrs.change_password**
  Description: Changes the password for a user.
  HTTP Method: POST
  Form Action: http://HOST:PORT/jrserver?jrs.cmd=jrs.change_password
  Content Type: application/x-www-form-urlencoded
  Content: jrs.cmd=jrs.change_password&jrs.uid=admin&jrs.password=CURRENT_PASSWORD&jrs.new_password=NEW_PASSWORD&jrs.confirm_new_password=NEW_PASSWORD
  Content Example: jrs.cmd=jrs.change_password&jrs.uid=admin&jrs.password=ad&jrs.new_password=1234&jrs.confirm_new_password=1234
- **jrs.get_preference_page**
  Description: Gets an HTML page in order to change the user preference settings with a web browser.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.get_preference_page
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.get_preference_page
- **jrs.logout**
  Description: Logs out from the JRServlet.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.logout
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.logout
- **jrs.login**
  Description: Logs into the JRServlet from a login dialog with web browsers or the JRViewer.
  HTTP Method: GET/POST
  URL Format: http://HOST:PORT/jrserver?jrs.cmd=jrs.login
  URL Example: http://localhost:8888/jrserver?jrs.cmd=jrs.login

Authentication parameters

JReport requires that the user be authenticated before running a report. If the user has not previously been authenticated a popup dialog requires you to give your user name and password before you can deal with the resources on the server. Under this circumstance, if you want to avoid the login dialog in the web browser, use either of the following two parameters:

- **jrs.authorization**
  Tag of the HTTP query field jrs.authorization.

  Description: Loads the JRServlet and avoids the login dialog in web browsers.
  Format of the value of the HTTP query field: Base64-encoded (userID:password).
  Example: If the user ID and password are both set as admin, then the value of the HTTP query field is Base64-encoded("admin:admin")="YWRtaW46YWRtaW4=", and the URL will be as follows: `http://localhost:8888/jrserver?jrs.cmd=jrs.get_subnodes&jrs.authorization=YWRtaW46YWRtaW4%3D`. For the Base64 encoding method, refer to http://en.wikipedia.org/wiki/Base64.

  Notes:

  - The JRServlet does not check security when an HTTP request has no jrs.cmd in the HTTP query in the root path of the JRServlet, and also does not accept the jrs.authorization for the request `http://localhost:8888/jrserver?jrs.authorization=YWRtaW46YWRtaW4%3D`.
  - This technique is not recommended for use outside of a firewall, although it looks like it is encrypted it is not secure.

- **jrs.auth_uid and jrs.auth_pwd**
  Tags of HTTP query field: jrs.auth_uid and jrs.auth_pwd.

  Description: Loads the JRServlet and avoids the login dialog in web browsers.
  Format of the value of the HTTP query field: jrs.auth_uid=USER_ID, jrs.auth_pwd=PASSWORD.
  Example: If the user ID and password are both set as admin, the URL will be as follows: `http://localhost:8888/jrserver?jrs.path=/&jrs.cmd=jrs.get_subnodes&jrs.auth_uid=admin&jrs.auth_pwd=admin`

  Note: The JRServlet does not check security when an HTTP request has no `jrs.cmd` in the HTTP query in the root path of the JRServlet, and also does not accept the jrs.auth_uid and jrs.auth_pwd for the request `http://localhost:8888/jrserver?jrs.auth_uid=admin&jrs.auth_pwd=admin`.

Accessing the report creation wizard via URL

Page reports and web reports use different URL entries and parameters for connecting to their own report creation wizards.

Accessing the page report wizard via URL

URL entry: /dhtmljsp/newreport.jsp

Parameters:

- jrs.catalog
  Description: The catalog name with full path.
- FromServer
  Description: Optional. The value is true and it cannot be changed.
- Authentication parameters: jrs.authorization, jrs.auth_uid, and jrs.auth_pwd
  Description: Optional.
- jrs.ds_name
  Description: Optional. The data source name in the specified catalog. If the data source is given (not null), only the business views in the data source will be shown.
- jrs.query_name
  Description: Optional. The query name in the specified catalog. If the query name is given (not null), the step of selecting business views in the report wizard will be skipped and will not be shown as a step. The business view based on the query will be used, and the jrs.cube_name property should be ignored.
- jrs.cube_name
  Description: Optional. The business view name in the specified catalog, that is the Name property value of the business view in the Catalog Browser of JReport Designer. If the business view name is given (not null), the step of selecting business views in the report wizard will be skipped and will not be shown as a step. If there are some business views with the same name, one of them will be used.
- jrs.param$
  Description: Optional.
- jrs.param_page
  Description: Optional. Value: true/false. Default value: true. It controls whether to show the parameter dialog for specifying parameter values that are not provided by jrs.param$ in the URL. Such parameter dialogs include the one displayed during report running and those brought by actions that lead to the change of parameters after the report is rendered in Page Report Studio, for example, the actions such as inserting a formula with a new parameter, running a master/detail report, and so on. However, the dialog displayed via Menu > Report > Change Parameters is not affected by this property, because the dialog is to display all the parameters of the report. When it is true and the value of a middle level cascading parameter (including old style cascading parameter) is provided, the parameter dialog only shows the lower level parameters. When it is false, no involved parameter dialog will pop up during the whole report running. The given parameter values and the default values of the other parameters will be used to run the report.

Examples:

- `http://localhost:8888/dhtmljsp/newreport.jsp?FromServer=true&jrs.catalog=/SampleReports/SampleReports.cat`
- `http://localhost:8888/dhtmljsp/newreport.jsp?FromServer=true&jrs.catalog=%2fSampleReports%2fSampleReports.cat`

You will then be directed to the New Page Report dialog which leads you to create a page report in the specified catalog.

Accessing the web report wizard via URL

URL entry: webos/app/webstudio/run.jsp?jrd_wizard=1&

All the parameters are encapsulated as JSON (JavaScript Object Notation) objects:

- jrd_catalog={
  "name":"xxx", // The catalog name with full path.
  "ver":"-1", // Optional. The catalog version. -1 means the latest version. Default value is "-1".
  "real":"false", // Optional. If you use absolute resource path, you need to add the property "real":"true" for the path. Default value is "false".
  "dsName":"xxx", // Optional. The data source name in the catalog. If the data source is given (not null), only the business views in this data source are shown in the report wizard.
  "QueryName":"xxx", // Optional. The query name in the catalog. If the query name is given (not null), the selecting business view drop-down list in the report wizard will be disabled. The business view based on the query will be used, and the BVName property below should be ignored.
  "BVName":"xxx", // Optional. The business view name in the catalog, that is the display name in the Catalog Browser of JReport Designer. If the business view name is given (not null), the selecting business view drop-down list in the report wizard will be disabled. If there are some business views with the same name, one of them will be used.
  "param_page":"true" // Optional. Value: true/false. Default value: true. It controls whether to show the parameter dialog displayed during report running for specifying parameter values that are not provided by jrd.param$ in the URL. The parameter panel in Web Report Studio is not affected by this property, because the dialog is to display all the parameters of the report. When it is true and the value of a middle level cascading parameter (including old style cascading parameter) is provided, the parameter dialog only shows the lower level parameters. When it is false, no parameter dialog will pop up and the given parameter values and the default values of the other parameters will be used to run the report.
  }
- jrd_param$={ // Optional.
  "p1":"v1", // p1 is parameter name, and v1 is p1's value.
  "p2":["v1","v2","v3"] // For multiple values.
  }
- Authentication parameters: jrs.authorization, jrs.auth_uid, and jrs.auth_pwd // Optional.

Examples:

- `http://localhost:8888/webos/app/webstudio/run.jsp?jrd_wizard=1&jrd_catalog={"name":"/SampleReports/SampleReports.cat","ver":"-1"}`
- Real path example: `http://localhost:8888/webos/app/webstudio/run.jsp?jrd_wizard=1&jrd_catalog={"name":"E:\\JReport\\Server\\jreports\\SampleReports\\SampleReports.cat","real":"true"}`

Running reports via URL

There are two ways you can use to run reports via URLs: JSP and servlet. Detailed JSP and servlet parameters are introduced in Appendix 6: Parameters for running reports via URL.

All the JSP and servlet methods introduced below are applicable to page reports. For web reports, only the server JSP and Servlet methods are applicable, while only tryView.jsp can be used to run web reports in Web Report Studio and the other methods can only be used to run web reports to other formats other than to Web Report Studio.

Using JSPs

Two server JSPs:

- **tryView.jsp**
  This is the normal method of accessing reports using URLs. If the report has parameters and you would like a parameter dialog to display for you to specify the parameter values, follow the topic Specifying parameter values using dialog when running or scheduling reports via URL, otherwise it will directly go to runReport.jsp.

  The tryView.jsp can be used to run page reports and web reports to any allowed formats.

  Examples:

  - Run a page report in Page Report Studio:
    `http://localhost:8888/jinfonet/tryView.jsp?jrs.cmd=jrs.try_vw&jrs.report=%2fSampleReports%2fEmployee Information.cls&jrs.catalog=%2fSampleReports%2fSampleReports.cat&jrs.result_type=8`
  - Run a web report in Web Report Studio:
    `http://localhost:8888/jinfonet/tryView.jsp?jrs.cmd=jrs.try_vw&jrs.report=/SampleReports/Link.wls&jrs.catalog=/SampleReports/SampleReports.cat&jrs.result_type=8`

- **runReport.jsp**
  The runReport.jsp functions the same as tryView.jsp when the report has no parameters. When it has parameters, the report runs with the default parameters if no parameter values are specified, or else it runs with the parameters specified in the URL.

  The runReport.jsp can be used to run page reports to any allowed formats and run web reports to formats other than in Web Report Studio.

  Examples:

  - Run a page report in Page Report Studio:
    `http://localhost:8888/jinfonet/runReport.jsp?jrs.cmd=jrs.web_vw&jrs.report=%2fSampleReports%2fEmployee Information.cls&jrs.catalog=%2fSampleReports%2fSampleReports.cat&jrs.result_type=8`
  - Run a web report to PDF:
    `http://localhost:8888/jinfonet/runReport.jsp?jrs.cmd=jrs.web_vw&jrs.report=%2fSampleReports%2fProducts.wls&jrs.catalog=%2fSampleReports%2fSampleReports.cat&jrs.result_type=2`

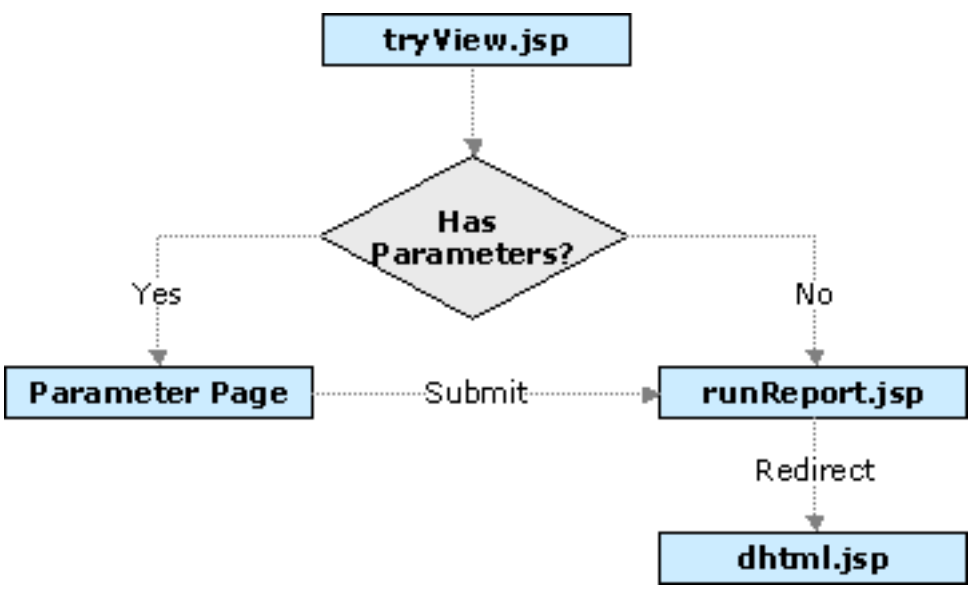
One Page Report Studio JSP:

- **dhtml.jsp**
  This JSP is for running page reports in Page Report Studio specifically, and you needn't specify the format type compared to the server JSPs. If you want to include the Page Report Studio JSPs into your own application without other server JSPs, you can use this method. However, you must specify the report parameter values in the URL if there are parameters used by the report. For example:

  `http://localhost:8888/dhtmljsp/dhtml.jsp?jrs.catalog=%2fSampleReports%2fSampleReports.cat&jrs.report=%2fSampleReports%2fEmployee Information.cls`

  > **Tip:** When you run a report in Page Report Studio using URL, if you want JReport to automatically refresh the report data at certain intervals, you can set the URL as follows:
  >
  > `http://localhost:8888/dhtmljsp/dhtml.jsp?jrs.report=/USERFOLDERPATH/admin/AutoRefresh/autorefresh.cls&jrs.catalog=/USERFOLDERPATH/admin/AutoRefresh/testauto.cat&jrs.auto_refresh_data=true&jrs.auto_refresh_data_time=10`

The image below illustrates the relationship between these JSPs:



Using servlet

Server servlet

The server servlet can be used to run page reports to any allowed formats and run web reports to formats other than in Web Report Studio.

Call the servlet *jrserver*. For example, run the page report in Page Report Studio:

`http://localhost:8888/jrserver/SampleReports/SampleReports.cat/Employee Information.cls?jrs.cmd=jrs.web_vw&jrs.result_type=8`

This request will be redirected to /dhtmljsp/dhtml.jsp with the following parameters:

jrs.catalog=/SampleReports/SampleReports.cat

jrs.report=/SampleReports/Employee Information.cls

Run a web report to HTML:

`http://localhost:8888/jrserver/SampleReports/SampleReports.cat/Link.wls?jrs.cmd=jrs.web_vw&jrs.result_type=1`

Page Report Studio servlet

The Page Report Studio servlet *jrdhtml* is used to run a page report in Page Report Studio specifically, and you needn't to specify the format type compared to the server servlet.

Examples:

- `http://localhost:8888/jrdhtml/SampleReports/SampleReports.cat/Employee Information.cls?`
- `http://localhost:8888/jrdhtml?jrs.catalog=/SampleReports/SampleReports.cat&jrs.report=/SampleReports/Employee Information.cls`

Tip: When running reports to the PDF format via URL, you can encrypt the PDF result by using the encrypting PDF properties.

Below are some more specific examples about running reports via URL:

- Running a report tab in a page report via URL
- Opening web reports in Web Report Studio via JSON
- Sending a large page report result by e-mail upon finishing
- Switching report database connection in URL
- Using the POST method

Related topics:

- Running dashboards via URL

Running a report tab in a page report via URL

If you want to run a specific page report tab, use jrs.report_sheet$RPT_TAB_NAME=true to specify a report tab in the current page report, where RPT_TAB_NAME is the report name of the specific report tab, not the display name. For example, jrs.report_sheet$Report2=true.

To get the report name and display name of a page report tab you can open the page report in JReport Designer and look at the Instance Name in the Properties panel or you can make use of the API methods getName() and getDisplayName() in the interface jet.server.api.ReportSheetInfo. For the detailed usages, see the JReport Javadoc located in `<install_root>\help\api`.

The URL for running a report tab Applet in Page Report Studio within the report MultimediaObjects.cls is as follows:

```
http://localhost:8888/jinfonet/tryView.jsp?jrs.cmd=jrs.try_vw&jrs.catalog=%
2fSampleReports%2fSampleReports.cat&jrs.report=%2fSampleReports%2fMultimediaObjects.
cls&jrs.result_type=8&jrs.report_sheet$Report1=true
```

Or

```
http://localhost:8888/jrserver/SampleReports/SampleReports.cat/MultimediaObjects.cls?
jrs.cmd=jrs.try_vw&jrs.result_type=8&jrs.report_sheet$Report1=true
```

Opening web reports in Web Report Studio via JSON

JReport provides parameters for developer users to run web reports in Web Report Studio via URL. All the parameters are encapsulated as JSON (JavaScript Object Notation) objects. Therefore, it will help if you obtain some knowledge on JSON to understand the syntax more clearly. When composing the URL, you need to use URL encoding to avoid errors.

The following lists the parameters for running web reports via a URL:

- jrd_report={
  "name":"xxx", // The full path of the web report.
  "ver":"-1", // Optional. The report version. -1 means the latest version.
  }
- jrd_catalog={
  "name":"xxx", // The full path of the catalog that the web report uses.
  "ver":"-1", // Optional. The catalog version. -1 means the latest version.
  }
- jrd_param$={
  "p1":"value1", // p1 is parameter name, and value1 is p1's value.
  "p2":["value1","value2","value3"] // For multiple values.
  }
- jrd_userinfo={ // Optional.
  "user":"xxx", // User name.
  "country":"us", // The locale representing the region part for running reports, following locale naming specification.
  "language":"en", // The locale representing the language part for running reports, following locale naming specification.
  "encoding":"UTF-8", // The encoding for running reports.
  "resolution":"96" // The resolution for displaying reports.
  }
- jrd_security_file_name={
  "name":"xxx", // The name of a security file. When a security file is given, the security definitions in the specified catalog will be replaced by that defined in the security file.
  // Optional.
  }
- jrd_datasources=[
  // Optional.
  // The following types of external data sources are supported:
  // 0 - JDBC data source, 1 - JNDI data source, 2 - Java DataSource object, 3 - Connection object, 4 - ResultSet object.
  // One or multiple data sources can be included at a time.

  {
  // JDBC data source.
  "ds":"Data Source 1", // Data source name.
  "uid":"xxx", // DB user name.
  "pwd":"xxx", // DB user password.
  "type":"0", // Indicates it is JDBC data source.
  "url":"xxx", // JDBC URL. For example, "url":"jdbc:oracle:thin:@127.0.0.1:1521:ora8i".
  "driver":"xxx" // JDBC driver. For example, "driver":"oracle.jdbc.driver.OracleDriver".

  // When connecting to a different database with different database metadata or data metadata, you need to also specify the following commands to set the target database metadata information if there are differences:
  "refresh_support_info":"True", // Optional. Value: True/False. If it is true, JReport will get support information including reverse words, quote characters, and so on from the driver each time fetching data from the database. It is recommended that you set this property to true when connecting to a different database.
  "quote_character":"xxx", // Optional. The quote characters.
  "extra_characters":"xxx", // Optional. The extra characters.
  "date_format":"xxx", // Optional. The date format.
  "datetime_format":"xxx", // Optional. The datetime format.
  "time_format":"xxx", // Optional. The time format.
  "transaction_readonly":"xxx", // Optional. The transaction "read only" property. Possible value: default, Read only, or Read&Write. Ignore other values.
  "transaction_mode":"xxx", // Optional. The transaction mode. Possible value: default, None, Read Uncommitted, Read Committed, Repeatable Read, or Serializable. Ignore other values.
  "char_to_be_replaced":"xxx", // Optional. The characters that are to be replaced.
  "char_replaced_by":"xxx" // Optional. The characters used to replace the characters specified by *char_to_be_replaced*.
  },

  {
  // JNDI data source. Currently JReport only supports local JNDI with anonymous lookup which means that the JNDI data source and the report server are in one JVM and the JNDI needs no authentication.
  "ds":"Data Source 2", // Data source name.
  "uid":"xxx", // DB user name.
  "pwd":"xxx", // DB user password.
  "type":"1", // Indicates it is JNDI data source.
  "url":"xxx", // JNDI name. For example, "url":"jndi/testDB".
  },

  {
  // Java DataSource object/Connection object/ResultSet object.
  // Users should define request or session attribute, then the attribute key is the one defined in the request.
  // For example, write user.jsp as follows:
  // String key = "Rst";
  // java.sql.ResultSet rst = null;
  // rst = // Gets result set object from user own business logic.
  // request.setAttribute("Rst", rst);
  // Then the key would be that "key":"Rst".
  // The above example is based on ResultSet object. It also applies to the other two types.
  "ds":"Data Source 3", // Data source name.
  "type":"2/3/4", // Indicates the data source type.
  "key":"xxx" // Request attribute object name, included in request or session.
  }

  ]

Here is an example of the complete URL without URL encoding to make it easier to read:

```
http://localhost:8888/webreport/studio/entry/run.jsp?jrd_report={"name":"/SampleReports/report.wls","ver":"-1"}&jrd_catalog={"name":"/SampleReports/SampleReports.cat","ver":"-1"}&jrd_param$={"P_Coutry":"USA"}&jrd_security_file_name={"name":"SampleReports.security.xml"}&jrd_datasources=[{"ds":"Data Source 1","uid":"xxx","pwd":"xxx","type":"0","url":"xxx","driver":"xxx"},{"ds":"Data Source 2","type":"2","key":"xxx"}]
```

If you use absolute resource path, you need to add the property "real":"true" for the path. For example,

```
jrd_report={"name":"C:\\JReport\\Server\\jreports\\SampleReports\\Products.wls","real":"true"}&jrd_catalog={"name":"C:\\JReport\\Server\\jreports\\SampleReports\\SampleReports.cat","real":"true"}
```

Run Products.wls in the Public Reports > SampleReports folder:

```
http://localhost:8888/webreport/studio/entry/run.jsp?jrd_report={"name":"/SampleReports/Products.wls"}&jrd_catalog={"name":"/SampleReports/SampleReports.cat"}
```

Sending a large page report result by e-mail upon finishing

Here two ways are provided to publish a large report result via URL: JSP and servlet. Take the following examples to publish a large report result by e-mail and to customize the message displayed in the web browser.

- **Using JSP**

  See JSP parameters.

  Use tryView.jsp to run a report without a parameter:

  ```
  http://localhost:8888/jinfonet/tryView.jsp?jrs.cmd=jrs.try_vw&jrs.catalog=%
  2fSampleReports%2fSampleReports.cat&jrs.report=%2fSampleReports%2fEmployee
  Information.cls&jrs.timeout_send_email=true&jrs.report_timeout=5&jrs.
  mailto=person@company.com&jrs.mailsubject=TaskForTimeoutSendEmail&jrs.
  result_type=1&jrs.mailcomments=IFTHEREPORTISFINISHEDTHERESULTWILLBESENT&jrs.
  mailfrom=person@company.com
  ```

  Use runReport.jsp to run a report with parameters:

  ```
  http://localhost:8888/jinfonet/runReport.jsp?jrs.cmd=jrs.web_vw&jrs.catalog=%
  2fSampleReports%2fSampleReports.cat&jrs.report=%2fSampleReports%2fCustomerAnalysis.
  cls&jrs.param$P_StartDate=01/01/2006&jrs.param$p_EndDate=12/31/2007&jrs.
  timeout_send_email=true&jrs.report_timeout=1&jrs.mailto=person@company.com&jrs.
  mailsubject=AboutTaskForTimeout&jrs.result_type=1
  ```

- **Using Servlet**

  See Servlet parameters.

  Use a servlet to run a report without a parameter:

  ```
  http://localhost:8888/jrserver/SampleReports/SampleReports.cat/Employee Information.
  cls?jrs.cmd=jrs.try_vw&jrs.timeout_send_email=true&jrs.report_timeout=1&jrs.
  mailto=person@company.com&jrs.mailsubject=AboutTaskForTimeoutSendEmail&jrs.
  result_type=1&jrs.mailcomments=COUNTRECORDSPERPAGE
  ```

  Use a servlet to run a report with a parameter:

  ```
  http://localhost:8888/jrserver/SampleReports/SampleReports.cat/CustomerAnalysis.cls?
  jrs.cmd=jrs.web_vw&jrs.param$P_StartDate=01/01/2006&jrs.param
  $p_EndDate=12/31/2007&jrs.timeout_send_email=true&jrs.report_timeout=1&jrs.
  mailto=person@company.com&jrs.mailsubject=TaskForTimeoutEmail&jrs.result_type=1&jrs.
  mailcomments=CUSTOMERORDER
  ```

- **Customize your message to be displayed in the web browser**

  You can set a message by setting the parameter jrs.timeout_sendmail_message.

  For example, you want to display message as follows:

  Running <report name> takes more than <Timeout> seconds.
  The subject is <mail subject> and has been sent to <mailto> from <mailfrom>
  It is a file whose type is <type>.

  Then you can set the parameter in URL:

  jrs.timeout_sendmail_message=Running {6} takes more than {0} seconds.<p>The subject is {2} and has been sent to {1} from {5}.<p>It is a file whose type is {3}.

  Where

  {0} - The report timeout
  {1} - mail to
  {2} - mail subject
  {3} - result type
  {4} - mail comment
  {5} - mail from
  {6} - Catalog name/report
  <p> - an Enter key

  Example

  ```
  http://localhost:8888/jrserver/SampleReports/SampleReports.cat/CustomerAnalysis.cls?
  jrs.cmd=jrs.web_vw&jrs.param$P_StartDate=01/01/2006&jrs.param
  $p_EndDate=12/31/2007&jrs.timeout_send_email=true&jrs.report_timeout=1&jrs.
  mailto=person@company.com&jrs.mailsubject=AboutTaskForTimeoutSendEmail&jrs.
  result_type=1&jrs.mailcomments=CustomerAnalysis&jrs.mailfrom=person@company.com&jrs.
  timeout_sendmail_message=...
  ```

  This {6} is a large report whose runtime is over {0} seconds.<p>The report is sent to {1} from {5}.<p>The subject of the mail is {2}.<p>Its type is {3}.

  Note: You should type the single quote sign " ' " twice if you use it in the message.

Switching report database connection in URL

To access reports via URL, you can switch the connection in the same database or between different databases at runtime. As a result, if the databases you want to switch between have the same structure, you will then be free from having to build another similar catalog. You can use the switch database commands to set the JDBC connection or to change the user name/password in order to connect to another database.

The following are database connection commands:

- **jrs.jdbc_url**
  Constant: jet.cs.util.APIConst.TAG_JDBC_URL
  Description: Specifies the new JDBC URL in the catalog to run a report.
- **jrs.jdbc_driver**
  Constant: jet.cs.util.APIConst.TAG_JDBC_DRIVER
  Description: Specifies the new JDBC driver in the catalog to run a report.
- **jrs.db_user**
  Constant: jet.cs.util.APIConst.TAG_DB_USER
  Description: Specifies the new database user ID.
- **jrs.db_pswd**
  Constant: jet.cs.util.APIConst.TAG_DB_PSWD
  Description: Specifies the new database password.

When connecting to a different database with different database metadata or data metadata, you need to also specify the following commands to set the target database metadata information if there are differences:

- **jrs.db_refresh_support_info**
  Constant: TAG_DS_REFRESH_SUPPORT_INFO
  Description: Optional. If it is true, JReport will get support information including reverse words, quote characters, and so on from the driver each time fetching data from the database. If it is not set, the property will be treated as false. It is recommended that you set this property to true when connecting to a different database.
- **jrs.db_quote_character**
  Constant: TAG_DS_QUOTE_CHARACTER
  Description: Optional. Specifies the quote characters.
- **jrs.db_extra_characters**
  Constant: TAG_DS_EXTRA_CHARACTERS
  Description: Optional. Specifies the extra characters.
- **jrs.db_date_format**
  Constant: TAG_DS_DATE_FORMAT
  Description: Optional. Specifies the date format.
- **jrs.db_datetime_format**
  Constant: TAG_DS_DATETIME_FORMAT
  Description: Optional. Specifies the datetime format.
- **jrs.db_time_format**
  Constant: TAG_DS_TIME_FORMAT
  Description: Optional. Specifies the time format.
- **jrs.db_transaction_readonly**
  Constant: TAG_DS_TRANSACTION_READONLY
  Description: Optional. Specifies the transaction "read only" property. Possible value: "default", "Read only", or "Read&Write". Ignore other values.
- **jrs.db_transaction_mode**
  Constant: TAG_DS_TRANSACTION_MODE
  Description: Optional. Specifies the transaction mode. Possible value: "default", "None", "Read Uncommitted", "Read Committed", "Repeatable Read", or "Serializable". Ignore other values.
- **jrs.db_char_to_be_replaced**
  Constant: TAG_DS_CHAR_TO_BE_REPLACED
  Description: Optional. Specifies the characters that are to be replaced.
- **jrs.db_char_replaced_by**
  Constant: TAG_DS_CHAR_REPLACED_BY
  Description: Optional. Specifies the characters used to replace the characters specified by *jrs.db_char_to_be_replaced*.

Switching the connection and user/password in the same database

- Set the Oracle database named oracle815 connection when designing the report Report1.cls, and later switch the connection to the Oracle database named demo at runtime.

  The URL for switching the connection:

  `http://localhost:8888/jrserver/SampleReports/SampleReports.cat/Report1.cls?jrs.cmd=jrs.try_vw&jrs.result_type=1&`**`jrs.jdbc_url`**`=jdbc:oracle:thin:@host:1521`**`:demo`**
- Set the SQL database named MBA2000 when designing the report Report1.cls, and later switch the connection to the SQL database named JTTest at runtime.

  The URL for switching the connection:

  `http://localhost:8888/jrserver/SampleReports/SampleReports.cat/Report1.cls?jrs.cmd=jrs.try_vw&jrs.result_type=1&`**`jrs.jdbc_url`**`=jdbc:inetdae:host:1433?database=`**`JTTest`**`&sql7=true`
- Specify the user ID system/manager to ensure security when designing the web report Report2.wls, and then switch to the user ID Scott and the password tiger when running the report in Web Report Studio.

  The URL for switching the user ID and password:

  `http://localhost:8888/jinfonet/tryView.jsp?jrs.cmd=jrs.try_vw&jrs.report=/SampleReports/Report2.wls&jrs.catalog=/SampleReports/SampleReports.cat&jrs.result_type=8&`**`jrs.db_user`**`=Scott&`**`jrs.db_pswd`**`=`**`tiger`**
- Set the Sybase 12 database named master when designing the web report Report2.wls, and later switch the connection to the Sybase 12 database named product at runtime.

  The URL for switching the connection:

  `http://localhost:8888/jrserver/SampleReports/SampleReports.cat/Report2.wls?jrs.cmd=jrs.try_vw&jrs.result_type=1&`**`jrs.jdbc_url`**`=jdbc:sybase:Tds:host:5000/`**`product`**

Switching the connection between different databases with the same database metadata

- Set oracle815 connection when designing the report Report1.cls, and then switch the connection to Access database with the JDBC-ODBC driver named products at runtime.

  The URL for switching the connection:

  `http://localhost:8888/jrserver/SampleReports/SampleReports.cat/Report1.cls?jrs.cmd=jrs.try_vw&jrs.result_type=1&`**`jrs.jdbc_driver`**`=sun.jdbc.odbc.JdbcOdbcDriver&`**`jrs.jdbc_url`**`=jdbc:odbc:`**`products`**
- Set oracle815 connection when designing the web report Report2.wls, and then switch the connection to SQL server database named products at runtime.

  The URL for switching the connection:

  `http://localhost:8888/jrserver/SampleReports/SampleReports.cat/Report2.wls?jrs.cmd=jrs.try_vw&jrs.result_type=1&`**`jrs.jdbc_driver`**`=com.inet.tds.TdsDriver&`**`jrs.jdbc_url`**`=jdbc:inetdae:JT_P05:1433?database=`**`products`**`&sql7=true`

Switching the connection between different databases with different database metadata

The URL for switching the database to Oracle when running the page report report1.cls:

```
http://localhost:8080/remote/sub/jinfonet/tryView.jsp?jrs.cmd=jrs.try_vw&jrs.catalog=/Test/Test.cat&jrs.report=/Test/report1.cls&jrs.cmd=jrs.try_vw&jrs.result_type=8&jrs.jdbc_driver=oracle.jdbc.driver.OracleDriver&jrs.jdbc_url=jdbc:oracle:thin:@192.168.0.1:1521:oracle9&jrs.db_user=test&jrs.db_pswd=1234&jrs.db_quote_character="&jrs.db_date_format=M/d/yyyy&jrs.db_datetime_format=d/M/yyyy h:mm:ss a&jrs.db_time_format=HH:mm:ss.SSS&jrs.db_transaction_readonly=Read Only&jrs.db_transaction_mode=Repeatable Read&jrs.db_char_to_be_replaced=}&jrs.db_char_replaced_by=}oracle91234
```

Related topics:

- Running reports via URL
- Appendix 6: Parameters for running reports via URL
- Opening web reports in Web Report Studio via JSON for switching connection for web reports via JSON
- Running dashboards via URL for switching connection for dashboards

Using the POST method

- **In HTML pages**

  Specify POST as the method of a form in an HTML page.

- **In Java program**

  For example:

```
URL url = new URL("http://jrserver:8888");
URLConnection uc = url.getConnection();
if (uc instanceof HttpURLConnection) {
  HttpURLConnection huc = (HttpURLConnection)uc;
  //set use POST method.
  huc.setRequestMethod("POST");
  huc.setRequestProperty("Content-Type", "application/x-www-form-urlencoded");
  huc.setDoOutput(true);

  //write the HTTP query to the output stream.
  OutputStreamWriter writer = new OutputStreamWriter(huc.getOutputStream());
  writer.write("jrs.cmd=jrs.get_subnodes");
  writer.close();
  huc.getHeaderField(0);

  //get the response content from the server.
  InputStream inStream = uc.getInputStream();
  if (inStream != null) {
    BufferedReader reader = new BufferedReader(new InputStreamReader(inStream));
    String inputLine;
    while (null != (inputLine = reader.readLine())) {
      System.out.println(inputLine);
    }
  }
}
```

Scheduling reports via URL

A schedule contains two kinds of properties. The first is about time information and the second is about task information (for detailed information about each property, see Appendix 7: Properties for scheduling reports via URL).

When reports are scheduled via URL in a web browser, the method jrs.submit_schedule is called. The following is a description of this method.

- **Constant:** jet.cs.util.APIConst.CMD_SUBMIT_SCHEDULE
- **Description:** Submits a scheduled task to JReport Server. If the report has no parameters then it directly run the URL. If the report has parameters and no parameter specified in the URL or the parameters provided in the URL fail to include all necessary parameters, the server then returns the parameter dialog for entering parameter values.
- **HTTP Method:** GET/POST.
- **Response:** Returns the scheduled tasks page to web browsers.

URL examples

The following are some specific examples on how to schedule a report via URL. The report CustomerAnalysis.cls that uses the catalog /SampleReports/SampleReports.cat is used in these examples.

Publishing to the versioning system immediately

To publish the report to the versioning system immediately, the URL used for this task could be as follows:

- **To RST**
  http://localhost:8888/jrserver?jrs.cmd=jrs.submit_schedule&jrs.catalog=%2fSampleReports%2fSampleReports.cat&jrs.task_class=jet.server.schedule.jrtasks.PublishRptTask&jrs.launch_type=0&jrs.param$P_StartDate=2006-01-01&jrs.param$p_EndDate=2007-12-31&jrs.uid=admin&jrs.to_version=true&jrs.to_version_rst=true&jrs.report=%2fSampleReports%2fCustomerAnalysis.cls
- **To PDF**
  http://localhost:8888/jinfonet/submitSchedPage.jsp?jrs.cmd=jrs.submit_schedule&jrs.catalog=%2fSampleReports%2fSampleReports.cat&jrs.task_class=jet.server.schedule.jrtasks.PublishRptTask&jrs.launch_type=0&jrs.param$P_StartDate=2006-01-01&jrs.param$p_EndDate=2007-12-31&jrs.uid=admin&jrs.to_version_pdf=true&jrs.to_version=true&jrs.report=%2fSampleReports%2fCustomerAnalysis.cls

If you want to publish the report to other formats, you can refer to the examples above, but you will need to modify the parameter jrs.to_version_pdf=true to suit the particular format that the report will be published to. For example, if you want to publish the report to Excel, you will need to change the parameter to jrs.to_version_excel=true.

Publishing to the versioning system periodically

To publish the report to the versioning system periodically, the URL used for this task could be as follows:

http://localhost:8888/jrserver?jrs.cmd=jrs.submit_schedule&jrs.catalog=%2fSampleReports%2fSampleReports.cat&jrs.hour2=5&jrs.task_class=jet.server.schedule.jrtasks.PublishRptTask&jrs.to_version=true&jrs.to_version_html=true&jrs.is_between=true&jrs.is_weekday=false&jrs.launch_type=8&jrs.min2=0&jrs.is_pm=false&jrs.hour=9&jrs.is_pm2=true&jrs.param$P_StartDate=2006-01-01&jrs.param$p_EndDate=2007-12-31&jrs.uid=admin&jrs.hours=1&jrs.is_hourly=true&jrs.at_min=0&jrs.days_id=0&jrs.day=1&jrs.timezone=CTT&jrs.min=0&jrs.expire_days=30&jrs.report=%2fSampleReports%2fCustomerAnalysis.cls&jrs.rpt_language=en&jrs.report_sheet$Report=true&jrs.schedule_name=periodicalTask

Publishing to the versioning system at a specific time

To publish the report to the versioning system at 12:00:00 CST, December 1, 2008, the URL used for this task could be as follows:

http://localhost:8888/jrserver?jrs.cmd=jrs.submit_schedule&jrs.catalog=%2fSampleReports%2fSampleReports.cat&jrs.task_class=jet.server.schedule.jrtasks.PublishRptTask&jrs.to_version=true&jrs.to_version_rst=true&jrs.launch_type=1&jrs.param$P_StartDate=2006-01-01&jrs.param$p_EndDate=2007-12-31&jrs.uid=admin&jrs.exe_day=1&jrs.exe_month=12&jrs.exe_hour=12&jrs.exe_min=00&jrs.exe_year=2008&jrs.expire_days=30&jrs.report=%2fSampleReports%2fCustomerAnalysis.cls&jrs.report_sheet$Report=true&jrs.schedule_name=atTimeTask

Publishing to the versioning system immediately and notifying others of success or failure

To publish the report to the versioning system immediately, and notify the user support1, support2, and support3 of success or failure, the URL used for this task could be as follows:

http://localhost:8888/jrserver?jrs.cmd=jrs.submit_schedule&jrs.catalog=%2fSampleReports%2fSampleReports.cat&jrs.task_class=jet.server.schedule.jrtasks.PublishRptTask&jrs.launch_type=0&jrs.param$P_StartDate=2006-01-01&jrs.param$p_EndDate=2007-12-31&jrs.uid=admin&jrs.expire_days=30&jrs.report=%2fSampleReports%2fCustomerAnalysis.cls&jrs.to_version=true&jrs.to_version_rst=true&jrs.success_notify=true&jrs.fail_notify=true&jrs.notification_emails=To%3A+person1@company.com%0D%0ACc%3A+person2@company.com%0D%0ABcc%3A+person3@company.com%0D%0A

Publishing to the file system immediately

To publish the report to the file system immediately, the URL used for this task could be as follows:

- **To RST**
  http://localhost:8888/jinfonet/submitSchedPage.jsp?jrs.cmd=jrs.submit_schedule&jrs.catalog=%2fSampleReports%2fSampleReports.cat&jrs.task_class=jet.server.schedule.jrtasks.PublishRptTask&jrs.launch_type=0&jrs.param$P_StartDate=2006-01-01&jrs.param$p_EndDate=2007-12-31&jrs.uid=admin&jrs.to_disk_rst_path_type=1&jrs.rst_dir=C:\&jrs.to_disk=true&jrs.to_rst=true&jrs.rst=CustomerAnalysis.rst&jrs.report=%2fSampleReports%2fCustomerAnalysis.cls&
- **To PDF**
  http://localhost:8888/jrserver?jrs.cmd=jrs.submit_schedule&jrs.catalog=%2fSampleReports%2fSampleReports.cat&jrs.task_class=jet.server.schedule.jrtasks.PublishRptTask&jrs.launch_type=0&jrs.param$P_StartDate=2006-01-01&jrs.param$p_EndDate=2007-12-31&jrs.uid=admin&jrs.to_disk_pdf_path_type=1&jrs.pdf_dir=C:\&jrs.to_disk=true&jrs.to_pdf=true&jrs.pdf=CustomerAnalysis.pdf&jrs.report=%2fSampleReports%2fCustomerAnalysis.cls&

Note: If you want to publish a report to a disk path, for example, jrs.rst_dir=C:\, you can do the same as the example described above. If you want to publish a report to a resource tree, for example, jrs.rst_dir=%2fSampleReports, you can omit this parameter &jrs.to_disk_pdf_path_type=1 in the above example and replace jrs.pdf_dir=C:\ with jrs.rst_dir=%2fSampleReports. Here SampleReports is the resource path.

Publishing to e-mail immediately

To publish the report to e-mail immediately, the URL used for this task could be as follows:

http://localhost:8888/jinfonet/submitSchedPage.jsp?jrs.cmd=jrs.submit_schedule&jrs.catalog=/SampleReports/SampleReports.cat&jrs.report=/SampleReports/CustomerAnalysis.cls&jrs.task_class=jet.server.schedule.jrtasks.PublishRptTask&jrs.to_mail=true&jrs.jrmail0=jrs.mailto%3dperson@company.com%26jrs.mailsubject%3dreport&jrs.launch_type=0&jrs.param$P_StartDate=2006-01-01&jrs.param$p_EndDate=2007-12-31&jrs.uid=admin&jrs.rpt_language=en

Publishing to printer immediately

To publish the report to a printer immediately, the URL used for this task could be as follows:

http://localhost:8888/jrserver?jrs.cmd=jrs.submit_schedule&jrs.catalog=%2fSampleReports%2fSampleReports.cat&jrs.task_class=jet.server.schedule.jrtasks.PublishRptTask&jrs.launch_type=0&jrs.to_printer=true&jrs.report=%2fSampleReports%2fCustomerAnalysis.cls&jrs.param$P_StartDate=2006-01-01&jrs.param$p_EndDate=2007-12-31&jrs.rpt_language=en

Publishing to fax immediately

To publish the report to a fax immediately, the URL used for this task could be as follows:

http://localhost:8888/jinfonet/submitSchedPage.jsp?jrs.cmd=jrs.submit_schedule&jrs.catalog=%2fSampleReports%2fSampleReports.cat&jrs.report=%2fSampleReports%2fCustomerAnalysis.cls&jrs.task_class=jet.server.schedule.jrtasks.PublishRptTask&jrs.launch_type=0&jrs.param$P_StartDate2006-01-01&jrs.param$p_EndDate=2007-12-31&jrs.uid=admin&jrs.to_fax=true&jrs.to_fax_is_include_cover=false&jrs.to_fax_to_fax_number=888888

Publishing to FTP immediately

To publish the report to an FTP site immediately, the URL used for this task could be as follows:

http://localhost:8888/jrserver?jrs.cmd=jrs.submit_schedule&jrs.catalog=%2fSampleReports%2fSampleReports.cat&jrs.report=%2fSampleReports%2fCustomerAnalysis.cls&jrs.task_class=jet.server.schedule.jrtasks.PublishRptTask&jrs.launch_type=0&jrs.param$P_StartDate=2006-01-01&jrs.param$p_EndDate=2007-12-31&jrs.uid=admin&jrs.to_FTP=true&jrs.ftp0=jrs.ftpHost%3d192.168.0.0%26jrs.ftpPort%3d21%26jrs.ftpUn%3dftpuser%26jrs.ftpPsd%3d1234%26jrs.ftpLoc%3d%2ftest%26jrs.ftpPdf%3dtrue

Publishing a bursting report to disk and to PDF

Parameters for bursting reports:

- jrs.is_bursting_task=true
  Description: Indicates it is to schedule a bursting report task.
- jrs.bursting_schema$Schema_Name=true
  Description: Specifies the names of the schemes to schedule.

The following URL publishes the bursting report SaleStatForRegion.cls to disk and to PDF:

http://localhost:8888/jinfonet/submitSchedPage.jsp?jrs.auth_uid=admin&jrs.auth_pwd=admin&jrs.cmd=jrs.submit_schedule&jrs.catalog=%2fSampleReports%2fSampleReports.cat&jrs.report=%2fSampleReports%2fSaleStatForRegion.cls&jrs.task_class=jet.server.schedule.jrtasks.PublishRptTask&jrs.launch_type=0&jrs.bursting_schema$BurstingSchema=true&jrs.is_bursting_task=true&jrs.to_disk=true&jrs.to_pdf=true&jrs.disk_pdf_encrypt=true&jrs.password_pdf=true&jrs.pdf_permis_pasw=1234&jrs.confirm_new_password=1234&jrs.pdf_printallow=2

Specifying parameter values using dialog when running or scheduling reports via URL

When running or scheduling a report that requires parameters via URL, you need to provide the parameter values in the URL string without any mistakes. You may find this difficult sometimes because it's not easy to remember all parameters in a report and you can not take advantage selecting values when bound with a DBMS value. However, JReport provides you with the JSP tryView.jsp for running reports and the special command jrs.submit_schedule for report scheduling, which enables you to display the Enter Parameter Values dialog to manage parameters.

The display of the parameter dialog requires:

- The report has parameters.
- For running the report, the URL calls tryView.jsp and sets `jrs.param_page`=`true` or "`param_page`":"`true`" (for web reports only). While for report scheduling, the URL contains the command jrs.submit_schedule.
- No parameter specified in the URL or the parameters provided in the URL fail to include all necessary parameters.

With the above three conditions fulfilled, the Enter Parameter Values dialog will be displayed right after you complete the URL in the address bar and make it run. After setting the values and clicking OK in the dialog, JReport Server will append the values to the URL string for you and execute the URL. However, if you click Cancel in the dialog, running the report will be cancelled.

Tip: If you want to specify parameters via dialog when running or scheduling reports via URL, the most effective way to display the parameter dialog is that you do not specify parameters at all in the URL.

Embedding JReport into Your Application with JavaScript APIs

With the JavaScript APIs provided by JReport, you can successfully embed page reports, web reports, and dashboards into your own application and then perform actions outside the report template. The implemented JavaScript APIs in external applications can be applied to open, close, export, print, navigate, refresh and save reports and dashboards without using Page Report Studio, Web Report Studio, or JDashboard.

Supported operations for page reports

- Open a report
- Close application
- Export a report via API or by accessing the Page Report Studio Export dialog
- Print a report via API or by accessing the Page Report Studio Print dialog
- Refresh data
- Save the current active report
- Page navigation

Supported operations for web reports

- Open a report
- Close application
- Export a report via API or by accessing the Web Report Studio Export dialog
- Print a report via API or by accessing the Web Report Studio Print dialog
- Refresh data
- Save the current active report

Supported operations for dashboards

- Open multiple dashboards and switch among them
- Close application
- Export a dashboard using the JDashboard Export dialog
- Print a dashboard using the JDashboard Print dialog
- Refresh data
- Save the current active dashboard

JReport Server provides two demos in the `<install_root>\public_html\webos\app\demo` folder:

- jreportapi-demo-rpt.html for embedding a page/web report
- jreportapi-demo-dsb.html for embedding dashboards

To run the demos:

1. Start your JReport Server and then open a demo file.
2. In the displayed web page, click the **Open xxx** option on the left menu to load the specified report or dashboards as customized in the API.

   For the dashboard demo, two dashboards are opened and their names are shown above the dashboard body. You can click the names to switch between the two dashboards.
3. Use the left menu to perform actions on the report or dashboards.
4. If the report or dashboards has defined web controls such as filter control and custom control, they can also work.
5. If there are links in the report or dashboards, after you click a link, a link path is displayed above the report body or dashboard body. You can then use the link path to go back to the previous step.

As for the detailed JavaScript APIs and methods of each API that JReport provides, refer to the API documentation in the `<install_root>\help\javascriptapi` folder.

Single sign on support

The following shows how to specify a user in a single sign on environment as compared to nomal login.

For common login, user name and password are set as:

user: "admin",
pass: "admin",

See the following example:

```
var server = {
url: "http://localhost:8888/jinfonet/tryView.jsp",
user: "admin",
pass: "admin",
jrd_prefer:{
    // For page report
    pagestudio:{
        feature_UserInfoBar:false,
        feature_ToolBar: false,
        feature_Toolbox: false,
        feature_DSOTree: false,
        feature_TOCTree: false,
        feature_PopupMenu: false,
        feature_ADHOC: false
    },

    // For web report
    wrptstudio:{
        viewMode:{
            hasToolbar: false,
            hasSideArea: false
        }
    }
},
jrd_studio_mode: "view",
"jrs.param_page": true
},
```

For single sign on, use *authorized_user:"user_name"* to specify the authorized user as follows:

```
var server = {
url: "http://localhost:8888/jinfonet/tryView.jsp",
authorized_user:"admin",
jrd_prefer:{
    // For page report
    pagestudio:{
        feature_UserInfoBar:false,
        feature_ToolBar: false,
        feature_Toolbox: false,
        feature_DSOTree: false,
        feature_TOCTree: false,
        feature_PopupMenu: false,
        feature_ADHOC: false
    },

    // For web report
    wrptstudio:{
        viewMode:{
            hasToolbar: false,
            hasSideArea: false
        }
    }
},
jrd_studio_mode: "view",
"jrs.param_page": true
},
```

Working with APIs

APIs provide a way to build JReport components and services into other applications, thus integrating JReport with existing platform infrastructure. This chapter is aimed at Java developers wanting to understand how JReport Server works, and how to write Java code that uses the API effectively.

The comprehensive Java doc matrix covering all the classes and methods of the API is available in `<install_root>\help\api`. This chapter of the user guide is an introduction and overview of the APIs so that the vastness of the Java docs will be easier to master.

The technical architecture section reviews specific implementation details about JReport Server. The technical view is different from the product architecture covered in another section of the user guide. This shows the programming elements and processes that exist, rather than the services they provide. The focus is on pointing out Java web application elements and Java servlets and Java programming in the JReport Server product. It introduces the various elements that make up the full JReport API.

The tour of the API section is a high level survey of the JReport Server API. The intent is to show boundaries of interest so that a developer can narrow down the set of APIs to look at when needing to use JReport Server for a particular task.

Beyond that are explanations of use cases, with examples of code.

- Technical architecture
- Tour of the Java API
- Java API for a servlet
- Java API for an application
- JReport Server API
- JReport Catalog API
- JReport Design API
- JReport Result Viewer Bean
- JReport Catalog Bean
- JReport Exit functions

•

Technical architecture

JReport Server is built with Java and uses standard Java and web technologies. The report server functionality is deployed in classes packaged into jar files that are accessed by class loading based on class path settings. Working web applications built on top of the JReport Server are JSP pages and servlets running in a web server's Servlet Container, using HTTP protocol, generating HTML and Java Script that is passed to the browser.

This section gives a picture of how things fit together and what technologies are used.

This provides a context for later detailed explanations about writing Java code to extend an application to use JReport Server.

Software components

When talking about JReport Server, there are three levels of software components to understand.

- JReport Server - software components

Machine configurations

There are a variety of configurations for setting up a network to use JReport Server. This section will discuss how JReport Server may fit into a variety of network layouts, starting with the simplest plain JReport Server installed on a local PC or server. The discussion continues with how it fits with an existing web application. Then more complex configurations are explained, that scale up and support large web applications.

- JReport Server - out of the box - ready to run
- JReport Server - with existing web application
- JReport Server - remote dedicated server
- JReport Server - cluster

•

JReport Server - software components

The JReport Server product is a combination of several software components.

- At the highest level, there are JSP files that generate interactive web pages that form the JReport Server web application.
- Below that are compiled servlets that supply a group of useful actions controlled by query parameters in the requesting HTTP Request.
- At the bottom level is the Java API. This is a library of Java classes and methods to be used by developers to access JReport Server functionality.

The Java API is at the heart of the JReport Server functionality but it is the higher level components that provide immediate value and use.

JReport Server has production-ready web pages for accessing reports

JReport Server provides a large set of **JSP-based web pages** that are ready for production. These provide standard operations that most applications need.

When these JSP pages are deployed to the website, a visitor can access a web page that lists the set of reports available to them. From there they can schedule and run reports, including entering parameters that control the purpose and scope of the report. They can use this web application to view the results, or direct the reports to a PDF file, an e-mail, or a variety of destinations.

These pages are available immediately. Full report functionality is available through these interactive web interface without writing any Java code or knowing anything about the low level Java API.

These pages are written by Jinfonet engineers using the JReport Server Java API. Since they are JSP pages with visible source, they can be studied for ways of using the Java API or they can be refactored for a local variation of the web page. Using the CSS files and images it is also very easy to private label the application.

Providing these JReport Server web pages in your application is as easy as adding links that lead to the JSP pages. The JSP code can be customized as needed to fit into the design of the application so that they appear seamlessly as part of the whole.

Any web server technology can be used to build an application that includes these JReport Server web pages, including .NET and static HTML.

JReport Server has production-ready servlets for accessing reports

JReport Server also provides a set of **compiled servlets** that perform commonly needed actions related to running and scheduling reports, and viewing results. These services are requested via HTTP requests, with a set of parameters in the URL that define and control the requested service. They are called with parameters that completely define their service so it is possible to use them as helper operations behind an interactive web page that gathers information from a visitor and posts to the servlet URL, with parameters telling the servlet what to do with any resulting output.

Accessing these components of JReport is done by building and sending an HTTP request to the web server with a URL that points to the servlet. This can be done by a desktop program that interacts with a user to build up the needed parameters. A front end for the servlets could be a static HTML web page that provides a form for a user to fill in with the needed parameters before doing a submit to the URL. The web application can dynamically build appropriate links and put them into a menu or other page link. Any type web technology, including .NET can be used to build up an appropriate URL to call up the action of the servlets. Running the servlets does not depend on using Java.

There are examples of how the servlets are called in the sample file TestURL.html which is located in `<instal_root>\help\samples\URLSamples`.

JReport Server has a large set of classes and methods in its Java API

JReport Server provides a library of Java classes that can be used by a Java program. This is the **Java API**, which gives the most complete and versatile way to access JReport functionality. In later architectural illustrations, this may be labeled JReport Server API.

An application developer can extend their application to provide report functionality by writing custom code to call the JReport Server Java API to perform any operations needed. This is the key aspect of JReport Server that makes it appealing for application developers who need reporting functionality resident within the same web server.

Looking at the Javadocs for the Java API is intimidating. There are hundreds of classes and thousands of methods.

A later section gives a high-level tour of the Java API, organized by functional area. This will introduce the Java API so it is easier to understand. It will explain how the classes are organized to provide specific types of functionality.

A developer only needs to understand the working set of classes and methods that relate to a particular application need. A developer does not need to know about every class or method documented in the Java API Javadocs.

•

JReport Server - out of the box

JReport Server is a ready-to-run product that appears at first as a web application that lets users run reports and view results.

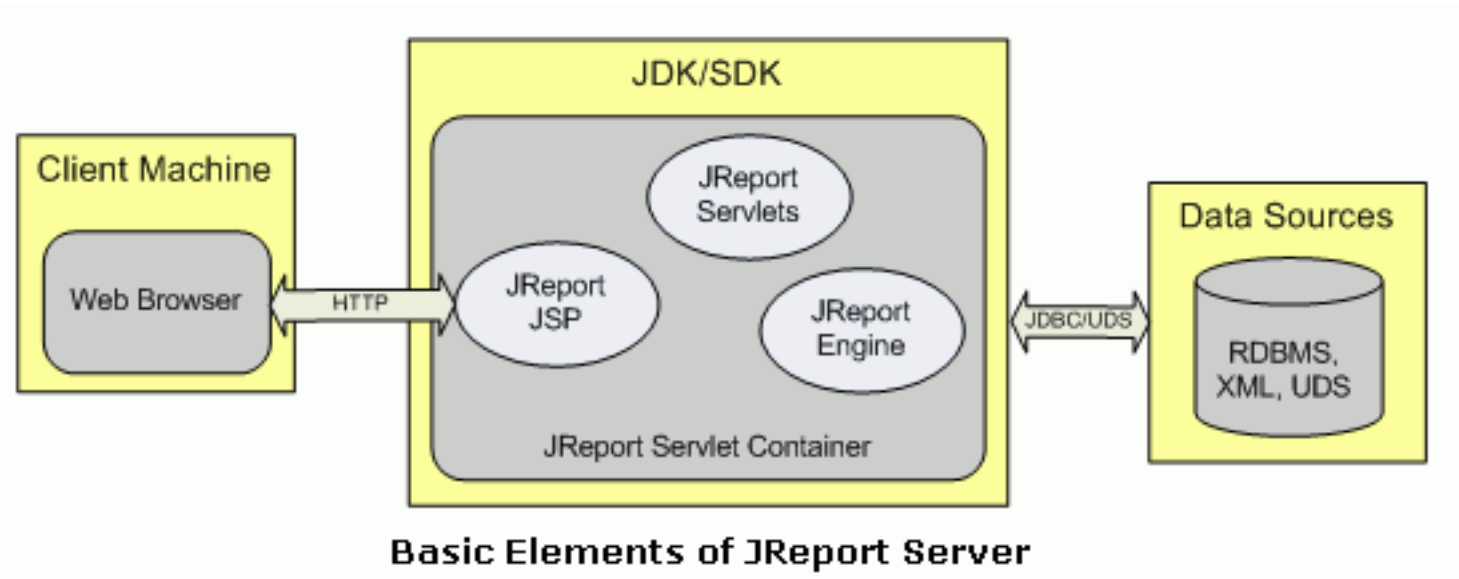
Install JReport Server and start it running by using the JRServer command or Windows Start menu. It is a web server ready for requests. It comes with a initial set of registered users and a set of sample reports that are in place to be run and viewed.

Start a browser and enter the URL to the entry page. Assuming the install is onto the local PC, this is the URL to use: `http://localhost:8888`.

Login with user/password of admin/admin. After login, the JReport Server web application is available.

What's running and how does it work?

JReport Server - web application



The JReport Server's web application is based on JSP files in several folders under the server's public_html folder and a few servlets compiled into classes. These JSP files and servlets create the HTML passed back to the browser, which gives users the web pages that access to the JReport services. A visitor to the site can see a list of reports available, select one to schedule or run, and view or download the results. This is a standard web application giving access to a set of sample report resources that come with the product.

Viewing report result may involve use of Java Script running within an HTML page. This javascript code is supplied by a library of files in the public_html\javascript folder.

The web application also includes a folder of JSP pages (public_html\admin) that are dedicated to performing administrative duties for managing and configuring the server. These can only be run by an administrative user.

The servlet code that generates web pages makes calls to classes in the JReport Engine which does the real work. These classes are the Java API.

The files that make up JReport Server's web application code are JSP files, compiled servlets, and compiled classes deployed in library jar files.

- <install_root>\public_html\jinfonet*
- <install_root>\public_html\dhtmljsp*
- <install_root>\public_html\admin*
- <install_root>\public_html\javascript*
- <install_root>\lib\JREntServer.jar
- <install_root>\lib\JREngine.jar
- <install_root>\lib\JREESServlets.jar

Login and servlet session

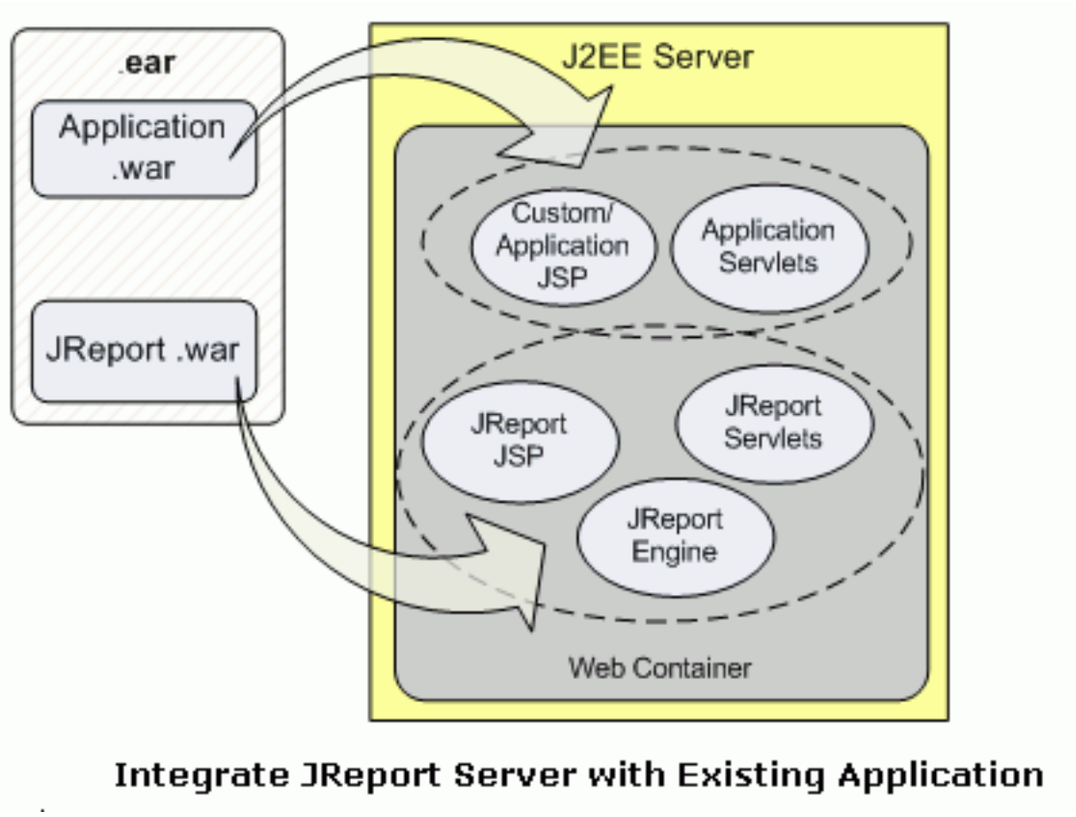
All web applications need to protect web page access from unauthorized use. JReport Server's web application includes a check at the start of every JSP page and servlet to verify that the requesting user is an authenticated JReport user before running.

See a detailed explanation of how this works in the Web page security section.

User identity for a web session is managed across HTTP requests by data in the servlet session that is passed into the Servlet container with the HTTP Request.

•

JReport Server - with existing web application



JReport Server can be integrated into an existing web application. Put the application's JSP files and servlets together with the JReport JSP files and servlets, along with the JReport Server library files. Read more about the details on how to do this in the Integrating with a Java Application Server chapter.

After JReport Server is installed along with the existing application, the application can be extended by writing new code that uses the JReport Server API library.

It is when JReport Server is used as part of a larger web application that it's other identity begins to show. JReport Server API comes into view. JReport Server contains is a large set of classes with methods that can be used by any Java program. This set of classes and methods is the JReport Java API.

Integrating JSP pages

An existing application can integrate with JReport Server web pages in a simple fashion by adding links from the existing application to the JReport Server JSP pages.

This provides the JReport Server web pages with the look and feel that is native to JReport Server.

Because the JReport Server web application is implemented as JSP pages, these can be edited to change the look and feel while not changing the functionality.

An existing application can be extended to access JReport Server functionality by adding Java code to access the Report Server classes and methods that are defined in the Java API. An existing application can have complete control over the look and feel of web pages while providing access to running and scheduling reports and viewing results.

Many of the JSP pages in JReport Server could be copied to the customer's application, modified, ignored or removed and have web interactions with users provided by an extended existing application, using Java calls to request report functionality.

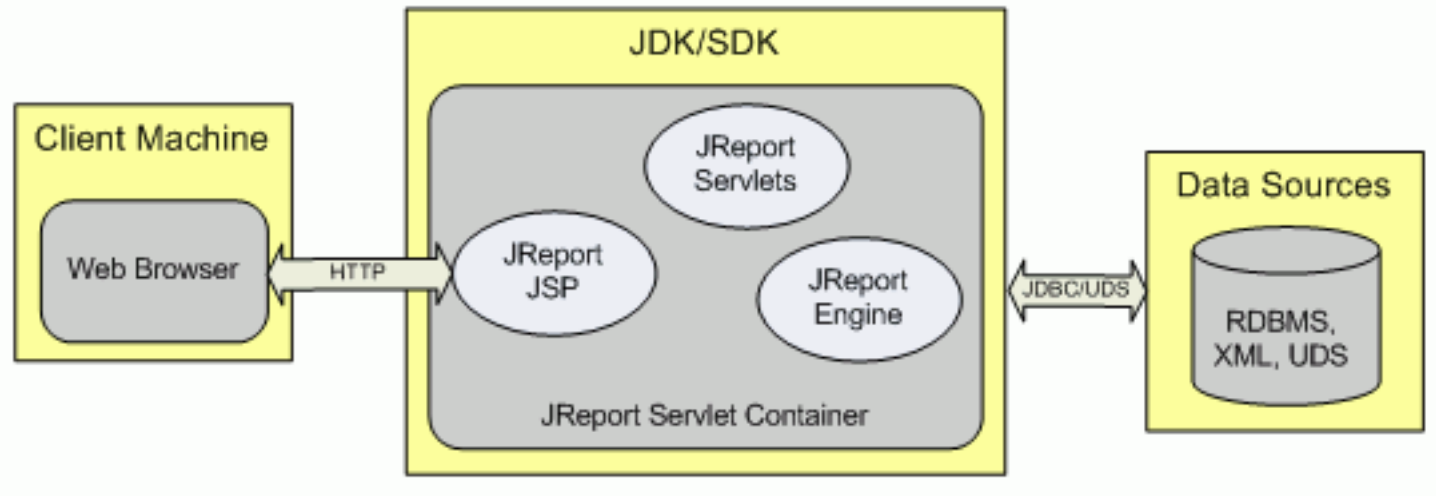
•

JReport Server - remote dedicated server

JReport Server can run in standalone mode directly as it is installed out of the box or can run as a Windows Service or as a WAR file running in any standard servlet container such as provided by Tomcat, JBoss, WebSphere, WebLogic, and so on. The customers application and users then access the JReport Server across the net either via RMI to servlets from Java application code or JSP page, or via HTTP requests with a URL that directly call JReport's servlets and JSP pages.

In a servlet container

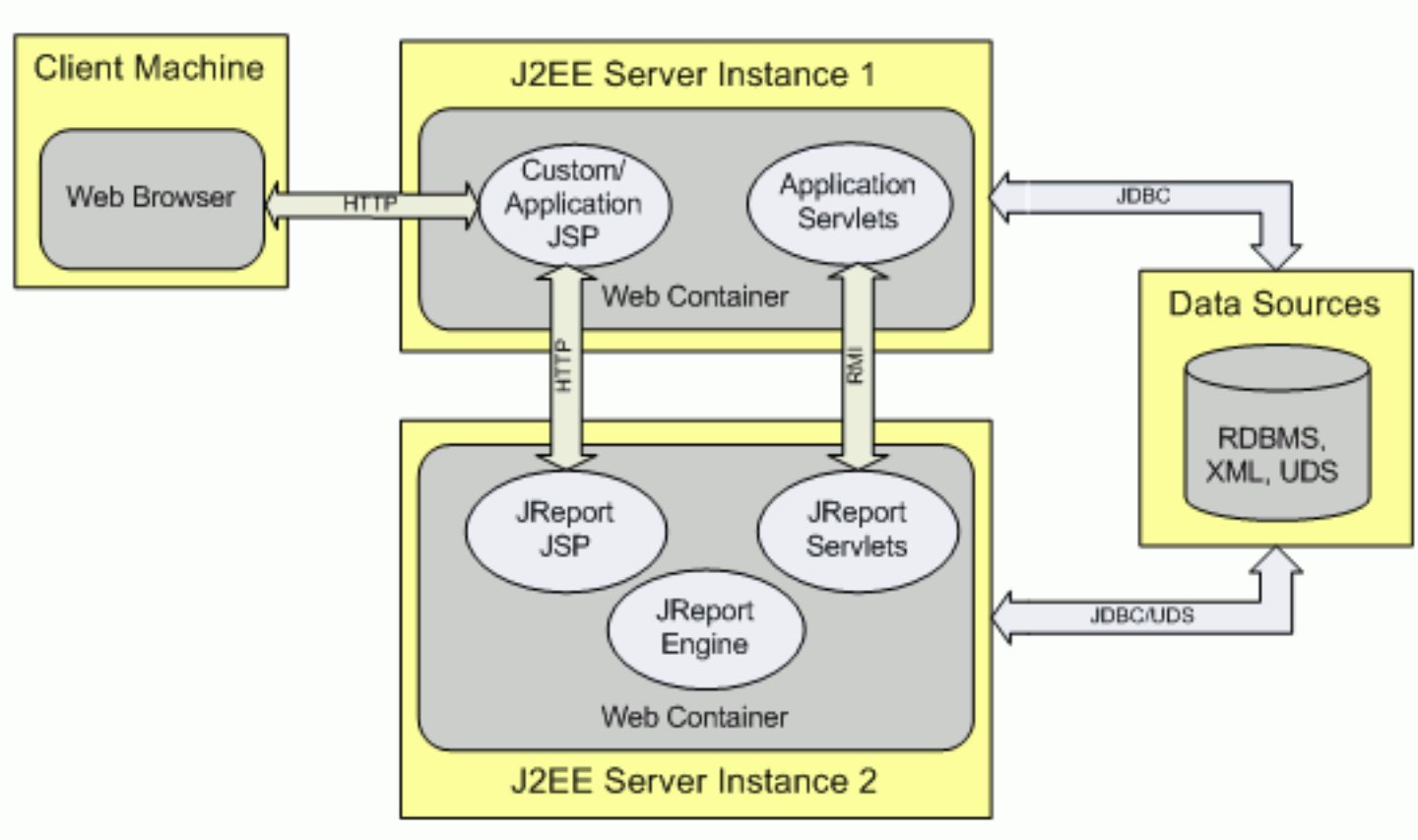
This is a common configuration for accessing JReport functionality for small applications. JReport Server is installed in a servlet container, ready to handle requests. The client machine can be simply a browser or be a full web application. It requests JReport functionality by using URLs to JReport servlets and JReport JSP pages running in the JReport Servlet Container. This is the enjoinment when you run JReport Server from the JRServer command line or build a jreport.war and and deploy it to an application server to run in it's own container.



JReport Server in a Servlet Container

As a remote dedicated server

When JReport is embedded into an existing application it is usually run in a separate JVM or a separate server machine. JReport runs as a back end server handling requests coming from the web application server's front end. JReport Servers' JSP pages are moved to the front end machine into the same servlet container or JVM as the application code. The front end web application's Java code runs with version of the Java API that are stubs providing seamless access to the back end methods using RMI (Remote Method Invocation) to call the real Java methods in the JReport Server in the other JVM.



JReport Server Accessed by RMI

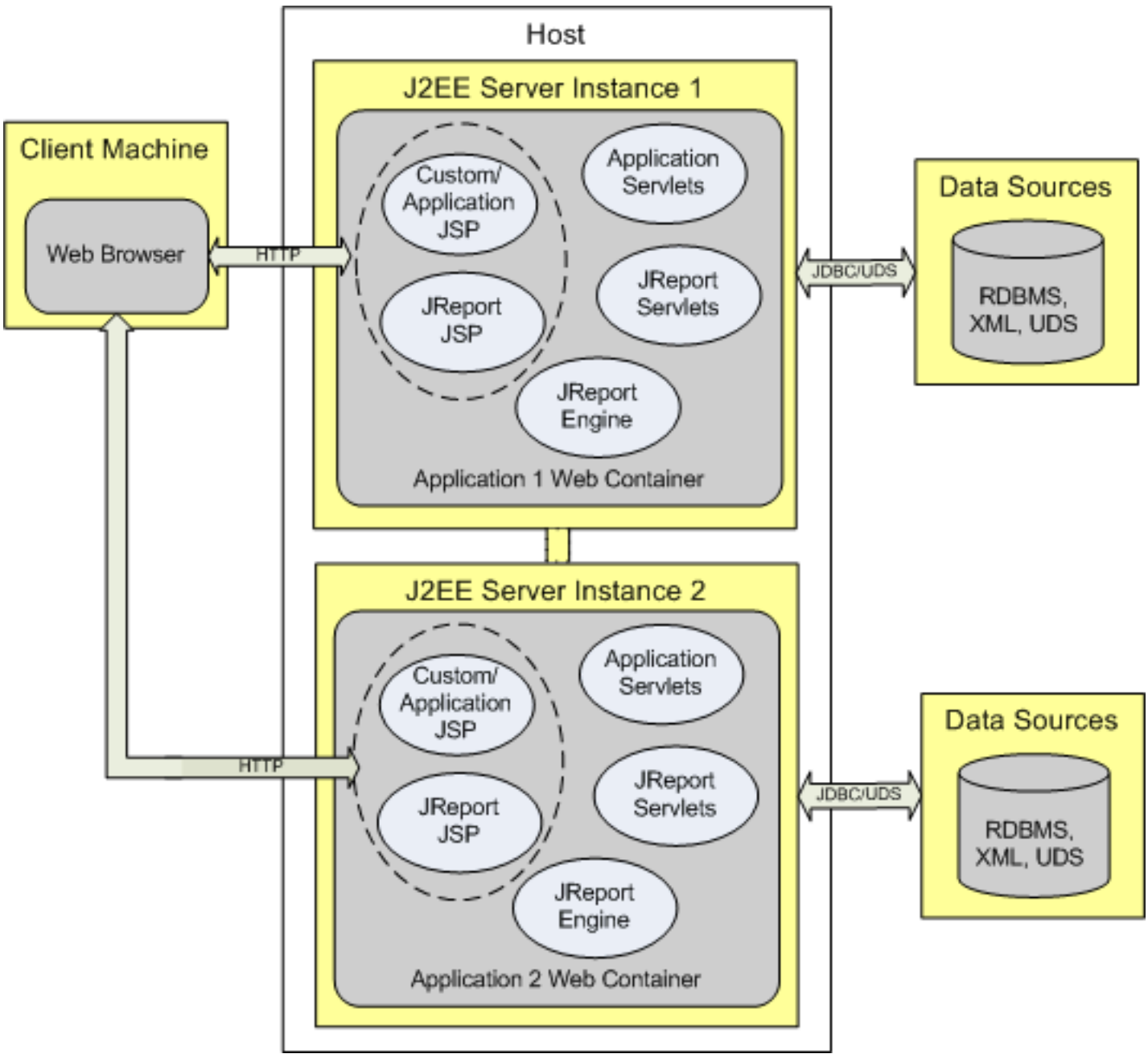
•

JReport Server - cluster

JReport can also run as a cluster using either a cluster of remote JReport standalone instances or as a cluster of JReport WAR files running inside a group of independent application server instances or as an application server cluster such as JBoss's cluster technology. When JReport is in a clustered environment, there is a built in dispatcher that has several available algorithms to distribute the runtime load. A Java interface is defined that allows application developers to write their own load distribution algorithm and register that with the JReport Cluster service.

JReport Server Cluster

The following diagram shows one potential configuration where JReport and the application are running in web containers. The JReport instances can be clustered so no matter where a resource is published it is visible on both systems and even if one system goes down for maintenance, the other system can handle the requests and new instances can be quickly started to handle larger loads. The client machines can go to either server so the user does not need to be aware of which JReport Server is handling the request.



Click here for more information about JReport Server Cluster.

Load Balancer

JReport provides several built in algorithms for load balancing and assigning tasks to servers. In most cases, one of the built in methods described in load balancing mechanism is adequate; however, JReport provides an interface called LoadBalancer that allows you to customize assigning tasks to specific server instances. For example, you may want to assign tasks to specific servers based on the user id yet still go to other servers if the specified server is overloaded or down.

You can view the Javadoc at jet.server.api.cluster.LoadBalancer and jet.server.api.rmi. RemoteDispatcher. See the examples in `<install_root>\help\samples\APICluster`.

Tour of the Java API

Using JReport Server with an existing application

Your application can use JReport Server with an application:

- Use the web application component of JReport Server JSP pages. Allow users to browse in the JReport Server web application to create reports and run reports. Modify the JReport Server JSP pages programs to offer special features based on calls the Java API. The web application components offer the easiest way to add JReport Server to your application, and provide the most flexible and powerful features and permissions. These can be used with an application built with any technology including .NET and HTML web pages.
- Use the Java API programs to make direct requests to URL. Although not technically an API, they serve the same purpose. Action requests are the method calls, with query parameters similar to method parameters. The set of server actions is limited to running, scheduling and viewing results. The URL can be called from any application. Results are returned. Use can request a URL including HTML and JReport Server will provide the Java web application.
- Use the Java API classes and methods. They are called directly from a Java program to extend functionality of JReport Server. The Java API provides the most flexibility and control. Use the Java API to create a custom web application, to create a custom scheduler and to add a custom JRP method. Use a Java API method if it does what you need. Use the Java API classes and methods to extend functionality.

Every function of JReport Server is available through classes in the Java API including creating and modifying catalogs and reports, providing security such as single sign on (SSO), administration and scheduling. Functions are available through the Java API that are not available in the web application.

Many Java API classes and methods are documented in the JReport Server Developer's Guide, providing and describing the method and parameters. It is this library of classes and methods that is explained below. Many Java API classes and methods are not documented in the Developer's Guide.

This tour section categorizes the Java API classes by functionality making it easier to find specific methods to use for particular functions.

Java Application Programming Interface (API)

The Java API provides the most flexible and powerful way to add JReport Server to your application.

The major functional areas in the library are categorized below into sections.

Each section provides a brief description of the class or interface name and have a general description. Below that will be a list of common methods, and other information as needed.

Initializing and connecting to JReport Server

Class: com.jreport.server.JReportServer

The JReportServer class provides methods for doing the initial connection and login. Use JReportServerParameter and query parameters to pass in log in credentials. The security mechanism includes a framework for single sign on (SSO) authentication. Use the JReportServerParameter class to pass in the SSO credentials to JReport Server. Use the JReportServerParameter class to pass in other parameters.

- **Control access to web pages**

Use JReport Server web application. This class provides a variety of utilities for programs running within a Java Servlet container.

- **Method:** getJReportServerInstance() - This method only returns when a validated JReport User is logged into JReport Server. Use this method to ensure that a user is logged in. If the user is not validated and login a user based on credentials in the current HTTP request. If no logged in user exists, then the login page is displayed. If a user is logged in, then the requested page is displayed. Cookies, authentication and the browser login dialog are used. The credentials are sent back in the same JSP page in the next HTTP Request.
- **Method:** getJReportServerInstance(HttpServletRequest) - This class provides methods that help to manage the JReport Server connection and login for web applications.
- **Method:** HttpUserSessionManager.check(login) - indicates if a validated JReport User is logged into the server. If a user is logged in, it returns true. If not, it returns false. If a login is supplied in the credentials in the current HTTP request, it does not automatically send a response to the browser to prompt for a login. Use getJReportServerInstance().

Authentication: Use the JReportServerParameter.setUsernamePassword() or the JReportServerParameter.setSSOToken() methods to pass in the credentials.

Single Sign On: Use JReportServerParameter.setSSOToken() - enter a print page for a group of JRP pages to be displayed. Use the JReportServerParameter.setUsernamePassword() - enter the login credentials for the JReport Server.

Network security: Using the check(login) system for connecting access to the web pages to JReport Server. Use the JReport Server web application. Use the JReport Server web application to create reports and run reports. Use the JReport Server web application security. Use JReport Server secure URLs to the web pages using HTTP (SSL) instead of HTTP. Secure Socket Layer (SSL) can easily be added to the JReport Server web application.

Authentication: Use the JReportServerParameter.setUsernamePassword() - enter the JReport User ID (user name) for the user who is currently logged into the web session.

Single Sign On: Use JReportServerParameter.setSSOToken() - enter a response to send to the browser redirecting the user to the application login page, and return a true to tell JReport Server to send the response to the browser. Use JReportServerParameter.setSSOToken().

Cookies: Use JReport Server web application. Use JReport Server web application security. Use implementation of the interface. Read comments in the source for detailed explanations on the implementation of the interface.

Authentication: Use JReportServerParameter.setUsernamePassword() - enter the JReport User ID (user name) for the user who is currently logged into the web session.

Single Sign On: Use JReportServerParameter.setSSOToken() - enter a print page for a group of JRP pages to be displayed. Use JReportServerParameter.setSSOToken().

Authentication: Use JReport Server web application security. Use JReport Server web application security. Use implementation of the interface. Read comments in the source for detailed explanations on the implementation of the interface.

Authentication: Use JReport Server web application security. Use JReport Server web application security. Use implementation of the interface. Read comments in the source for detailed explanations on the implementation of the interface.

Authentication: Use JReport Server web application security. Use JReport Server web application security. Use implementation of the interface. Read comments in the source for detailed explanations on the implementation of the interface.

**Single Sign On

Java API for a servlet

The Java servlet API is an JNI-based API for writing extensions to a web application or to a web server. In a JVM, this API is provided by the Java Servlet Extension (JSE) module. The API is also available in the Java Enterprise Edition (JEE) environment.

Servlets are Java classes that implement the Servlet interface. Most servlets are HTTP servlets, which implement the HttpServlet interface. Servlets are loaded into a web application by the JVM. The JVM then calls the servlet's methods to handle Java processing during HTTP transactions. Servlets can also be used to handle Java processing during HTTP transactions.

Servlets are loaded into a web application by the JVM. The JVM then calls the servlet's methods to handle Java processing during HTTP transactions.

Some Java Servlet technology provides two basic operations needed for sessions. Processing the requesting URL and forming the response URL. Servlet technology provides two methods for these operations. The first method is used for URL rewriting. The second method is used for URL encoding. Servlet technology provides a session tracking service that is used by servlets when processing a web page. Project Glassfish includes

Project GlassFish Servlet

```
Project GlassFish Servlet
```

Implementation

```
// Get service data factory object to get security parameter access
// Map of the form to the URL
ServiceDataFactory sdf = ServiceDataFactory.getInstance();
String[] ssmap = (String[]) getParameter("ssmap");
// Get the service data factory object to get the security parameter
// Map of the form to the URL
ServiceDataFactory sdf = ServiceDataFactory.getInstance();
```

•

Java API for an application

JReport provides two ways for the application to call JReport directly without using JSP pages and URLs.

- JReport Server API: JReport provides a library of methods to directly start JReport Server as part of the customers application running in the same JVM as the application.
- JReport Remote Server API: JReport provides methods to connect to an already running instance of JReport Server either running on the local machine in a different JVM or running on an entirely different server. Other than the first call to connect either using RMI or directly in the current JVM, the rest of the API is identical so it is easy to write an application that can work either remotely or locally the same as the JSP pages that JReport provides.

How to write client code using the Server API

If you want the entire JReport Server to run in the context of your application in the same JVM you can use the Server API. The Server when started this way starts a number of threads and runs exactly the same as when the server is started as a standalone server. The server just starts with the application and stops with the application. All access to the server such as RMI access from other applications is still allowed.

```
      //set report home
  System.getProperties().put("reporthome", "C:\\JReport\\Server");

      // Creates instance of RptServer ( this does nothing if JReport Server is already running in this
JVM)
  HttpUtil.initEnv(System.getProperties());

      // Get a handle to the server instance
  RptServer server = HttpUtil.getHttpRptServer();
```

How to write client code using the Remote Server API

It is exactly the same code as above except you get a remote handle to the RPTServer.

```
      //set report home for logging and location of rmi.auth file for security
  System.getProperties().put("reporthome", "C:\\JReport\\Server");
  System.getProperties().put("jrs.rmi.auth_file", "C:\\JReport\\Server\\bin\\rmi.auth");

      //search the running server using default host and port
  String host = "localhost";
  String port = "1129";

  RptServer server = RemoteReportServerToolkit.getRemoteWrappedRptServer(host, port);
```

JReport Server API

JReport Server API is a set of Java programming interfaces that run reports, explore report resources, and provide access control for report servers. It is mainly used for writing servlets, JSPs and Java applications with report server features.

The following topics are covered in this chapter:

- Installing the Server API
- Creating and getting instances of the ReportEngine
- Creating and getting instances of RptServer or HttpRptServer
- Using JSP page with a dedicated machine
- Using RMI in JReport Server
- Loading user data source classes at runtime
- Applying a user defined CSS to the HTML result file
- Specifying paths for the result files when using On-Demand API
- Setting dynamic parameter values when running/scheduling report via API
- Setting values of multi-value parameters via API
- Specifying report running properties in the session
- NLS API
- Dashboard Listener API
- Dynamic connection API
- Dynamic security API
- Information Bus API
- API demos
- RMI demos

Installing the Server API

When you install JReport Server, JReport Server API is also installed at the same time. After installation, you will have the following files:

- In `<install_root>\lib`:
  - JRESServlets.jar (local calls)
  - JREngine.jar (local calls)
  - JRSRMI.jar (remote RMI calls)
  - servlet.jar
  - hsqldb.jar
  - log4j-1.2.8.jar
  - sac.jar
  - commons-codec-1.2.jar
  - commons-logging-1.1.jar
  - quartz-all-1.6.5.jar
- In `<install_root>\derby\lib`:
  - derby.jar
  - derbyclient.jar
  - derbynet.jar
  - derbytools.jar

The Server API classes are stored in the archive file JRESServlets.jar.

When you install JReport Designer, the libraries for JReport Server can also be found in `<designer_install_root>\server\lib` since JReport Designer includes a full JReport Server for previewing reports.

If you are calling the Server API directly in the same JVM as JReport Server, you need to set the classpath environment variable to the local versions of the classes. Append the following jar files to your class path that compile and run applications which call the Server API:

```
<install_root>\lib\JRESServlets.jar; <install_root>\lib\JREngine.jar; <install_root>\lib
\servlet.jar; <install_root>\lib\hsqldb.jar; <install_root>\lib\log4j-1.2.8.jar;
<install_root>\lib\sac.jar; <install_root>\lib\commons-codec-1.2.jar; <install_root>\lib
\commons-logging-1.1.jar; <install_root>\lib\quartz-all-1.6.5.jar; <install_root>\derby
\lib\derby.jar; <install_root>\derby\lib\derbyclient.jar; <install_root>\derby\lib
\derbynet.ja; <install_root>\derby\lib\derbytools.jar;
```

If you are calling the Server API using RMI from a different JVM or different server than where JReport Server is running, you need to set the classpath environment variable to the remote libraries. Append the following jar files to your class path that compile and run applications which call the Server API:

```
<install_root>\lib\JRSRMI.jar; <install_root>\lib\servlet.jar; <install_root>\lib
\hsqldb.jar; <install_root>\lib\log4j-1.2.8.jar; <install_root>\lib\sac.jar;
<install_root>\lib\commons-codec-1.2.jar; <install_root>\lib\commons-logging-1.1.jar;
<install_root>\lib\quartz-all-1.6.5.jar;
```

Notes:

- If you are not using Derby as your Server DBMS, you need to replace the Derby libraries with the JDBC driver for your DBMS system.
- chart.jar is used in HTML applet and should be placed after jrengine.jar in the class path so as to avoid exceptions.
- If you want to export reports to the following formats, you should add the corresponding class package or jar with a valid path to the class path:
  - To e-mail or use the E-mail Notification function: activation-1.1.jar and mail-1.4.jar.
  - To FTP: commons-net-ftp-2.0.0.jar.
  - To PDF & Page Report Result: itext_1.5.4.jar.
  - To Excel: poiHSSF_151.jar.
  - To XML: xml-apis.jar and xercesImpl.jar.
  - To Page Report Result: JRWebDesign.jar.

Creating and getting instances of ReportEngine

To get an instance of the ReportEngine, you can use the method getInstance() or getInstance(boolean setDebugLevel) in the jet.server.api.engine.ReportEngineFactory.

For example,

```
bean = ReportEngineFactory.getInstance();
```

or,

```
bean = ReportEngineFactory.getInstance(true);
```

Reference: See JReport Javadoc in `<install_root>\help\api` for usage of the ReportEngine methods.

Creating and getting instances of RptServer or HttpRptServer

You can use the following methods to get access to the instance of JReport Server. JReport Server is a singleton - only one instance can exist in the system. Access to the server is through a returned RptServer object or a returned HttpRptServer object, depending on the environment of the request to get access. RptServer is designed to be instantiated in a Java Application while HttpRptServer is designed to be instantiated in a JSP page. HttpRptServer has a few extra methods available to use in the HTTP session.

Method 1

Use the method jet.server.api.http.HttpUtil.initEnv(Properties props).

The method jet.server.api.http.HttpUtil.initEnv(...) creates and initializes the HttpRptServer object. If the HttpRptServer has already been started in this application, it will use the existing HttpRptServer instance. Use this method to avoid creating more than one HttpRptServer instance.

```
        //  prepare report server initial properties.
        //  set report home minimally.
        //  System.getProperties() will bring in all configured settings.
    System.getProperties().put("reporthome", "C:\\JReport\\Server");

        //creates instance of ReportServer or does nothing if one is running already in this JVM
    HttpUtil.initEnv(System.getProperties());

    HttpRptServer server = HttpUtil.getHttpRptServer();

    // do something using the returned server object.
```

Method 2

Call the method HttpUtil.checkLogin(HttpServletRequest req, HttpServletResponse res).

The method jet.server.api.http.HttpUtil.checkLogin(...) implicitly calls HttpUtil.initEnv(...) if necessary. So you can get the HttpRptServer after calling HttpUtil.checkLogin(...).

```
        // use checkLogin to control access to rest of code in JSP file.
    if (HttpUtil.checkLogin(request, response))
        {
            // here, when a JReport user is logged into the current session.

            // get the instance of HttpRptServer .
        jet.server.api.http.HttpRptServer httpRptServer = HttpUtil.getHttpRptServer();

        // do something using the returned httpRptServer object.

        }
```

Using JSP with a dedicated machine

JReport Server can be configured so that JSP pages run on a web server machine while the JReport Server Engine is deployed to another machine. In this configuration, the web server machine is viewed as the client machine accessing the JReport Server system.

When this configuration is used, the Java API library used on the client side is different than the one used when JSP pages run in the same JVM as JReport Server. The method calls are the same, but the library for the client provides stubs for the methods that do a Remote Method Invocation (RMI) to the remote JReport Server to have the processing performed over there. All of the classes that are needed are in `<install_root>\lib\JRSRMI.jar`.

The use of RMI as the way to obtain processing is transparent to the JSP code in the client. Developers write exactly the same code in the JSP file. At deploy time, that JSP code will be bound to the proper library to use for the specific configuration.

By using JSPs in your web applications, the Remote Server API enables you to perform report running tasks remotely, to view the report result directly from the client side, and to administer JReport Server, without having to install JReport Server on the client machine.

The Report Server API is provided by the class jet.server.api.rmi. This class can be used explicitly to call methods that are specific to the configuration where RMI is used.

You can find more information about the jet.server.api.rmi package in the JReport Javadoc located in `<install_root>\help\api`.

Before using the Remote Server API, you must make sure that the machines have been configured properly:

1. Open the RMI service by configuration on the JReport Server side.

   Set the two properties server.rmiserver.enable and server.rmiadminservice.enable to **true** in the server.properties file located in `<install_ root>\bin`.

   When JReport Server resides behind a firewall, you need to specify a fixed port in order to pass through the firewall and obtain the remote objects from the client side by setting the server.rmiserver.fixed_port property in the server.properties file.

2. JReport Server remote host and port information is passed to the client application.

   Set the following parameters as the JVM environment variables in the client side:

   - Djrs.remote.host=<*HOST_NAME/IP_*address>
   - Djrs.remote.rmiport=<*HOST_PORT*>
   - Djrs.rmi.auth_file=<authFileName> (optional)

   You have two alternatives for setting these parameters:

   - You can add -Djrs.remote.host, -Djrs.remote.rmiport and -Djrs.rmi.auth_file (optional) to the batch file that is used to start your application.
   - If you are in an integration environment, you can also add the following to the web.xml file:

```
<context-param>
    <param-name>jrs.remote.host</param-name>
    <param-value>127.0.0.1</param-value>
</context-param>
<context-param>
    <param-name>jrs.remote.rmiport</param-name>
    <param-value>1129</param-value>
</context-param>
<!-- The third param(jrs.rmi.auth_file) is optional. "authFileName" includes the auth
file's realpath and auth file name. For example: C:\JReport\Server\bin\rmi.auth-->
<context-param>
    <param-name>jrs.rmi.auth_file</param-name>
    <param-value>C:\JReport\Server\bin\rmi.auth</param-value>
</context-param>
<listener>
    <listener-class>jet.server.servlets.JRServerContextListener</listener-class>
</listener>
```

Administering JReport Server remotely

JReport Server can be administered from a remote client by Remote Admin Service API. The administration of a JReport Server includes cluster administration, resource administration, security administration, configuration, connection pool management, resource alias management, connection information provider service, and catalog information management.

You can find the Remote Admin Service API in the jet.server.api.rmi.admin package available in the JReport Javadoc which is located in `<install_root>\help\api`.

Before a JReport Server can be performed the remote administration operation on, you need to set the following properties in the server.properties file located in `<install_ root>\bin` in the JReport Server side:

- **server.rmiserver.enable=true**
  Used to open the RMI service.
- **server.rmiadminservice.enable=true**
  Used to open the administration service in the RMI server. By default this property is set to false and the JReport Server does not support remote administration.
- **server.rmiserver.fixed_port=<AnOpenedPort>**
  Required when JReport Server runs behind a firewall.

Using single sign-on in RMI client

When using a Single Sign On (SSO) system, the web application machine is changed. This is where the work is done to set up the Single Sign On system and where to deploy the needed class implementations of the Java interface ExternalAuthorized.

For implementation of SSO in a standalone server, follow the steps below:

1. Write your HttpExternalAuthorized implementation. For details, see Single Sign On.
2. Compile Java classes. Compiling requires the library JRESServlets.jar, which can be found in `<install_root>\lib` directory.
3. Modify the class path used by your RMI client to include your external authentication classes, so that your application of RMI can access them.

   If it is in an integration environment, you can add the authentication classes in a folder named classes in the WEB-INF folder.
4. Modify the start file of your RMI client to define the system property jrs.httpExternalAuthorized with your implementation.

   For example, assuming that the implementation of this interface is com.mycorp.HttpExternalAuthorizedImpl.class, use the command line to start the RMI client:

   ```
   java -Djrs.httpExternalAuthorized=com.mycorp.HttpExternalAuthorizedImpl ....
   ```

   Then, restart the RMI client.

Using RMI in JReport Server

Remote Method Invocation (RMI) is used in JReport Server for distributed functionality, such as for clustering, event system, Remote API, and monitoring system. To avoid security risks and the mess of using the RMI registry (all the remote objects bound with the RMI registry directly can cause the remote objects to be unmanageable), **universal remote object management** has been introduced into JReport Server.

In the universal remote object management system, when getting a remote object to perform certain operations, either on a server (local or remote) or in a 3rd-party application (via Remote API), a request must be sent to the remote object management to get that special remote object. Also, authentication is required before any request for a remote object can be processed.

The authentication file

The authentication process requires a remote authentication file that contains the authentication information to accomplish the security check. The remote authentication file (`<install_root>\bin\rmi.auth`) is generated by JReport Server automatically during the server installation process. You can protect your authentication file by moving it to a safe place after installation.

Using the authentication file

The authentication file is commonly used in the following circumstances:

- In JReport Server, the remote object management system will use an authentication file, which resides in the `<server_install_root>\bin` directory.
- In JReport Server Cluster, make sure that the authentication files on all clustered servers are the same. If you want to add a new JReport Server that has a different authentication file to the server cluster, you should first backup the authentication file of that server, and then copy the authentication file from a clustered server in the JReport Server Cluster to `<server_install_root>\bin`.
- This file can then be used in JReport Server Monitor. Copy the authentication file generated by the JReport Server installation process to the directory `<monitor_install_root>\bin`. JReport Server Monitor will then fetch it from `<monitor_install_root>\bin` and build an authInfo object.
- If the Remote API is used, you must specify the authentication file when launching an application that contains the Remote API invocations. For example, `java -cp ... -Djrs.rmi.auth_file=%authFileName% mainClass`. Also, you can specify a string by specifying the content of the authentication file. For example, `java -cp ... -Djrs.rmi.auth_string=my_auth_info mainClass`. Alternatively, you can specify the authentication string yourself arbitrarily. As long as the authentication strings that the local and remote server both hold are the same, you can pass the authentication check: `java -cp ... -Djrs.rmi.auth_string=my_auth_info mainClass`.

  > **Note:** In JReport Server and JReport Server Monitor, you can also use -Djrs.rmi.auth_file or -Djrs.rmi.auth_string in the startup file. JReport Server/JReport Server Monitor looks up these three locations in turn (-Djrs.rmi.auth_string => -Djrs.rmi.auth_file => `<monitor_install_root>\bin`) when checking authentication information. The information first found will be used for the authentication check in the remote object management system.

Ways to generate authentication information

The authentication information can be of any bytes, as long as the authInfo objects match between the local and the remote server. The following are the methods for specifying authentication information:

- You can get the authentication file (rmi.auth) generated when installing the server.
- You can use the RMIAuthFileCreator.bat file stored in `<server_install_root>\bin` to generate the authentication file.
- You can compose any string into the auth file using any text editor.
- You can compose any string as the value of the system property -Djrs.rmi.auth_file or -Djrs.rmi.auth to specify authentication information.

Loading user data source classes at runtime

JReport allows you to create your reports using user data source (UDS), user defined objects (UDO), and user defined formulas (UDF). When you publish the reports using UDS, UDO or UDF to your production environment, JReport requires the UDS, UDO or UDF jars to be in the class path or part of the EAR bundle. This might be difficult for you. You need to shut down your application and wrap the new UDS, UDO or UDF classes into the WAR or EAR or put them in the class path of the application server.

The feature Loading User Classes at Runtime can solve this issue. That is, for standalone JReport Server, or in your embedded application server calling the Server API, you can control loading the classes at runtime without shutting down your application.

Methods

There are two important methods in the Server API that you need to call to dynamically load your UDS classes:

- **String jet.server.api.ServerEnv.getDynamicClassDir()**
  Retrieves the path of the dynamic classes of the report server.
- **static void jet.server.api.engine.ReportEngineFactory.loadClasses(String path)**
  Loads the dynamic classes from the location specified by the path parameter.

Usage

Follow the steps below:

1. Specify a folder location on the machine where JReport Server runs to hold all the UDS/UDO/UDF jar files, so that you can specify to load the classes from that folder. When JReport Server initializes, you can specify the location in one of the following ways:
   - **Using the Server API static void jet.server.api.http.HttpUtil.initEnv(java.util.Properties props)**

     Use the key server.dynamic.class.dir to set the location, for example:

     ```
     System.getProperties().put("server.dynamic.class.dir", "c:\\JReport\\Server\
     \dynamic"); HttpUtil.initEnv(System.getProperties());
     ```

   - **Using the option -Dserver.dynamic.class.dir in the server / application startup file**

     It will be loaded to system properties. Here is an example:

     `-Dserver.dynamic.class.dir=`*`YOUR_FOLDER`*

   - **Using the property entry server.dynamic.class.dir in the server.properties file located in <install_root>\bin**

     The default value of this property is `<install_root>\dynamicclasses`. Here is an example:

     `server.dynamic.class.dir=`*`YOUR_FOLDER`*.

     Note: In general, among the above three approaches, the first has the highest priority, and the third has the lowest.
2. Check the dynamic UDS/UDO/UDF folder any time by calling the method static String jet.server.api.ServerEnv.getDynamicClassDir().

   Usage of the method:

   ```
   HttpUtil.getHttpRptServer().getServerEnv().getDynamicClassDir()
   ```

   Example:

   ```
   System.out.println("Dynamic path"+httpRptServer.getServerEnv().getDynamicClassDir
   ());
   ```

   You can get the path's value in your own properties file. You can even copy your UDS/UDO/UDF jar files to the above folders programmatically.
3. Specify to load classes at runtime by calling the method static void jet.server.api.engine.ReportEngineFactory.loadClasses(java.lang.String path). The parameter path is the dynamic class folder.

   For example:

   ```
   ReportEngineFactory.loadClasses(httpRptServer.getServerEnv().getDynamicClassDir());
   ```

Notes (technical issues):

- Only *.jar or *.zip packages in the specified folder including resource files (recommended *.jar) will be loaded, which excludes the sub directories and unpackaged classes.
- If there are duplicated classes (having the same package and class name) in different *.jar files, the default loading sequence is in time order. The latest modified file will be loaded first, and the older ones will be neglected.
- If you split one dynamic UDS classes into different *.jar or *.zip packages and also want to take advantage of dynamic loader, then you should follow the below rules:
  - The jar with the main class implementing JRUserDataSource class of the dynamic UDS must be in the dynamic class folder.
  - The classes in the *.jar or *.zip packages in the dynamic class folder can call or refer the classes under the JVM's class path, it cannot work in reverse.
- If your UDS classes also go to load the recourse files,
  - You should bundle the resource files into a jar no matter whether it is in the same class jar file or not.
  - In addition, the classes (in the *.jar or *.zip) in the dynamic class folder can get the recourse files both in the dynamic class folder and under the JVM class path. However, the classes under the JVM class path cannot get the resource files in the dynamic class folder.
- When you call the API for loading the classes, JReport Server will not kill the running catalogs and reports but run them using the old UDS classes, and freeze new reports until the loading action has finished.
- JReport Server will try loading dynamic classes in the above way. If failed, it will try loading them in the traditional way (using the class path in the JReport Server startup file, or EAR/EAR in embedded integration environment). Once it fails to find the classes, ClassNotFoundException will be thrown.
- For the UDO classes, only the creator class and render class can be loaded.

Applying a user defined CSS to HTML result file

When exporting a report to HTML format, a .css file will be generated automatically by JReport to control the appearance and layout of the result file. When exporting a report to HTML format with Server API, JReport enables you to apply your own .css file to the HTML result file.

To apply user defined CSS to HTML result file:

1. In JReport Designer, specify the CSS selector in your own CSS for report objects.

   Select the object which you want to apply in your own CSS, and in Report Inspector, specify a value for the CSS property External CSS Class Selector. The value of the property should be the selector in your .css file.

2. Publish the report to JReport Server (for details, see Publishing resources).

3. Edit your JSP file that invokes the interface used to export the report to HTML format.

4. Export the report to HTML format using JReport Server.

JReport provides a demo JSP file ApplyUserCSS.jsp which is stored in `<install_root>\help\samples\JSPSamples\ApplyUserCSS` for your reference. You can copy the whole ApplyUserCSS folder to `<install_root>\public_html\jinfonet`, then

access the JSP via `http://host:port/jinfonet/ApplyUserCSS/ApplyUserCSS.jsp`.

Specifying paths for the result files when using On-Demand API

When you run a report in the Advanced mode using the on-demand report execution API, you can specify the path for the report result files. See the following demo code:

Demo code of running the report to the PDF format directly

```
//Set report properties
Properties props = new Properties();

//Set result type as .pdf format.
props.put(APIConst.TAG_RESULT_TYPE, String.valueOf(APIConst.PDF));

//Set result file location. The value of TAG_LOCATION_TO_SERVER_RESOURCE_PATH is 0,
//and the value of TAG_LOCATION_TO_SERVER_DISK_REAL_PATH is 1.
//The default value of TAG_RESULT_LOCATION_TYPE is 1.

//props.put(APIConst.TAG_RESULT_LOCATION_TYPE, String.valueOf(APIConst.
TAG_LOCATION_TO_SERVER_RESOURCE_PATH));
//props.put(APIConst.TAG_RESULT_LOCATION, "/");

props.put(APIConst.TAG_RESULT_LOCATION_TYPE, String.valueOf(APIConst.
TAG_LOCATION_TO_SERVER_DISK_REAL_PATH));
props.put(APIConst.TAG_RESULT_LOCATION, "C:\\");
System.out.println("========to pdf=" + props);
tempResult = rptServer.runReport(user, catalog, rptName, props);
```

Demo code of exporting the RST file to HTML format

```
//Set task properties
Properties props = new Properties();
props.put(APIConst.TAG_RESULT_TYPE, String.valueOf(APIConst.RST));
//Set result file location.
props.put(APIConst.TAG_RESULT_LOCATION_TYPE, "0");
props.put(APIConst.TAG_RESULT_LOCATION, "/");
System.out.println("========to pdf=" + props);

String tempResult = server.runReport("admin", catalog, rptName, props);
System.out.println("========tempResult="+tempResult);

//Export to HTML
props.put(APIConst.TAG_HTML, "Crosstab");
props.put(APIConst.TAG_HTML_DIR, "/");
props.put(APIConst.TAG_HTML_DIR_TYPE, String.valueOf(APIConst.TAG_LOCATION_TO_SERVER_RESOURCE_PATH));
System.out.println("\n========exportResult="+props);
ExportedFileList fileList = server.exportResult("admin", tempResult, props);
System.out.println("\n========fileList.getHTMLFilenameList()="+fileList.getHTMLFilenameList());
```

Setting dynamic parameter values when running/scheduling report via API

When running or scheduling a report with parameters via API, you can specify dynamic parameter values by implementing the ParameterGenerator interface. See JReport Javadoc in `<install_root>\help\api` for detailed usage of the API.

To set dynamic parameter values to a report:

1. Create a class MyParameterGeneratorImpl to implement the ParameterGenerator interface. See the API demo DemoParameterGenerator.java in `<install_root>\help\samples\APIParameter` for an example of implementing the ParameterGenerator interface.

2. Add the class path into the class path of your implementation of running or scheduling the report such as DemoParameterRunReport.

3. In DemoParameterRunReport, replace the way of setting static parameter values and instead set dynamic values of the required parameters in the report. Take the parameter PToday for example:

   First disable or delete the line:

   ```
   props.put(APIConst.TAG_PARAM_PREFIX + "PToday", "2007-5-21");
   ```

   and then add a line as follows:

   ```
   props.put(APIConst.TAG_PARAM_PREFIX + "PToday", APIConst.TAG_PARAM_GEN_PREFIX +
   "MyParameterGeneratorImpl";
   ```

Setting values of multi-value parameters via API

To run a report with multi-value parameters via API, take the following example to set values to a multi-value parameter Customers_Country:

```
String[] a = {"USA", "Canada"};
props.put(APIConst.TAG_PARAM_PREFIX + "Customers_Country", a);
```

If you want the parameter to use all its values, take the following:

```
props.put(APIConst.TAG_PARAM_PREFIX + "Customers_Country", new String[]{APIConst.
MULTIPLE_ALL_VALUE});
```

For the API code of running a report, see APIDemoRunReport.java in `<install_root>\help\samples\APIServer`.

Specifying report running properties in the session

When designing integration and security with your application, compared to putting all information in the URL, setting sensitive information in the session and letting the server fetch them when running reports is a more secure way.

JReport provides a key word JReport _running_parameter for holding property variables in the session. Users can set a string with this key word in the session before calling JReport services. The string should take the query string format used in URLs of running reports/dashboards and obey application/x-www-form-urlencoded format and be encoded by UTF-8.

When running reports or dashboards, JReport gets information first from URL and then picks up provided information from the session using the key words. For information that exists in both the URL and the session, the session setting overrides the URL settings.

For running reports via server JSP

For properties both in the URL and the session, the properties in the session will override those in the URL.

For properties only in the session, the properties will be appended to the URL to run the report.

Examples:

- **Set parameter value in the session:**

  HttpSession.setAttribute("JReport_running_parameter", "jrs.param$parameter1=xxx&jrs.param$parameter2=xxx...");

- **Set dynamic connection in the session:**

  HttpSession.setAttribute("JReport_running_parameter", "jrs.jdbc_driver=com.mysql.jdbc.Driver&jrs.jdbc_url=jdbc:mysql://IP:3306/test&jrs.db_user=xxx&jrs.db_pswd=xxx");

For running web reports via JSON

The following rules apply when merging each object from the session and from the URL:

- **jrd_report**
  Session overrides URL. Once this object is specified in the session, only the specified report can be run whenever the URL submits to run any report.

- **jrd_catalog**
  Session overrides URL.

- **jrd_param$**
  Session overrides the same-name parameters and appends the other parameters to the URL.

- **jrd_userinfo**
  Session overrides the properties in the object.

- **jrd_datasources**
  It should contain an object array and use "ds" as the key. The session overrides the objects with the same key and appends the other objects to the array.

Examples:

- **Set parameter value in the session (JSON format):**

  HttpSession.setAttribute("JReport_running_parameter", "jrd_param$={\"parameter1\":\"xxx\", \"parameter2\":\"xxx\"}");

- **Set dynamic connection in the session (JSON format):**

  HttpSession.setAttribute("JReport_running_parameter", "jrd_datasources=[{\"ds\":\"Data Source 1\", \"uid\":\"xxx\", \"pwd\":\"xxx\", \"type\":\"0\", \"url\":\"jdbc:mysql://IP:3306/test\", \"driver\":\"com.mysql.jdbc.Driver\"}]");

For running dashboards via JSON

The following rules apply when merging each object from the session and from the URL:

- **jrd_resext**
  For single properties, session overrides the properties in the URL and adds those not in the URL.

- **reslst**
  It should contain an object array and use "name" as the key. The objects with the same key plus dsh_params and dsh_datasources settings in the session will be appended to the object in the URL. Other objects from the session will be added to the array in the URL.

  For example:

  jrd_resext={"active":0,"reslst":[{"name":"/USERFOLDERPATH/admin/Dashboard1.dsh","dsh_params":[{"jrd_params":{"P_StartDate":"01/01/2006"}}]}]}

  Dashboard1.dsh will run with the parameter P_StartDate="01/01/2006".

  A user can set "JReport_running_parameter" in the session as:

  jrd_resext={"active":0,"reslst":[{"name":"/USERFOLDERPATH/admin/Dashboard1.dsh","dsh_params":[{"jrd_params": {"P_StartDate":"01/01/2007","P_EndDate":"12/31/2007"}}]}]}

  Dashboard1.dsh will then run with P_StartDate="01/01/2007" (applies the value setting from the session) and P_EndDate="12/31/2007" (append the other parameters from the session to the URL) for all library components.

Examples:

- **Set parameter value in the session for a specific library component:**

  HttpSession.setAttribute("JReport_running_parameter", "jrd_resext={\"active\":0,\"reslst\":[{\"name\":\"/62309_Session/62309.dsh\", \"dsh_params\":[{\"jrd_params\":{\"parameter1\":\"xxx\", \"parameter2\":\"xxx\"}}, {\"lc_names\":[\"s13dfc53ea67\"],\"jrd_params\":{\"parameter1\":\"xxx\",\"parameter2\":\"xxx\"}}, {\"lc_names\":[\"s13dfda08c0a\"],\"jrd_params\":{\"parameter1\":\"xxx\", parameter2:\"xxx\"}}]}]}");

- **Set dynamic connection in the session for specific library components:**

  HttpSession.setAttribute("JReport_running_parameter", "jrd_resext={\"active\":0,\"reslst\":[{\"name\":\"/62309_Session/62309_datasource.dsh\", \"dsh_datasources\":[{\"jrd_datasources\":[{\"ds\":\"Data Source 2\", \"uid\":\"root\", \"pwd\":\"1234\", \"type\":\"0\", \"url\":\"jdbc:mysql://IP1:3306/test\", \"driver\":\"com.mysql.jdbc.Driver\"}]}, {\"lc_names\":[\"/COMPONENT_LIB/62309_Session/62309_datasource.lc\"], \"jrd_datasources\":[{\"ds\":\"Data Source 2\", \"uid\":\"root\", \"pwd\":\"1234\", \"type\":\"0\", \"url\":\"jdbc:mysql://IP2:3306/test\", \"driver\":\"com.mysql.jdbc.Driver\"}]}, {\"lc_names\":[\"s13e0cdc04f2\"], \"jrd_datasources\":[{\"ds\":\"Data Source 2\", \"uid\":\"root\", \"pwd\":\"1234\", \"type\":\"0\", \"url\":\"jdbc:mysql://IP3:3306/test\", \"driver\":\"com.mysql.jdbc.Driver\"}]}]}]}");

NLS API

NLS API enables you to run reports in different languages, which includes global NLS API, catalog level NLS API, report level NLS API and resource tree NLS API. For detailed usages about the NLS API, see the JReport Javadoc in `<install_root>\help\api`.

Global NLS API

With the global NLS API, you can add a language, then specify to translate text in another language into a version in this language, or specify to replace a font face and size with another face and size. Likewise, you can get, update, or remove a language, a text translation, or a font replacement.

All text translations and font replacements specified for the global NLS will be applied each time a report runs.

Text is classified into four groups. You will find the four types in `NLSType` in the API Javadoc.

- COLUMN
  Text for columns. This type is for reports running in Page Report Studio.
- LABEL
  Text for labels and web controls.
- PROMPT
  Text for prompt information.
- TOC
  Text for resources displayed in the TOC tree.

For local environment

- Use this interface: `public interface RptServer`.
- Invoke `GlobalNLSLibrary getGlobalNLSLibrary()`.
- Call the methods that can help with your purposes.

For RMI usage

- Use this interface: `public interface RemoteRptServer`.
- Invoke `RemoteGlobalNLSLibrary getGlobalNLSLibrary()`.
- Call the methods that can help with your purposes.

Catalog level NLS API

With the catalog level NLS API, you can translate object names in a catalog to different languages.

For local environment

- Use this interface: `public interface NLSLibraryManager`.
- Invoke `CatalogNLSLibrary getCatalogNLSLibrary()`.
- Call the methods that can help with your purposes.

For RMI usage

- Use this interface: `public interface RemoteNLSLibraryManager`.
- Invoke `RemoteCatalogNLSLibrary getCatalogNLSLibrary()`.
- Call the methods that can help with your purposes.

Report level NLS API

With the report level NLS API, you can translate object names in a report to different languages.

For local environment

- Use this interface: `public interface NLSLibraryManager`.
- Invoke `ReportNLSLibrary getReportNLSLibrary()`.
- Call the methods that can help with your purposes.

For RMI usage

- Use this interface: `public interface RemoteNLSLibraryManager`.
- Invoke `RemoteReportNLSLibrary getReportNLSLibrary()`.
- Call the methods that can help with your purposes.

Resource tree NLS API

With the resource tree NLS API, you can translate object names in the resource tree of JReport Server to different languages.

For local environment

- Use this interface: `public interface RptServer`.
- Invoke `ResourceNLSManager getResourceNLSManager()`.
- Call the methods that can help with your purposes.

For RMI usage

- Use this interface: `public interface RemoteRptServer`.
- Invoke `RemoteResourceNLSManager getResourceNLSManager()`.
- Call the methods that can help with your purposes.

Dashboard Listener API

Dashboard Listener API defines the start and end events of adding and deleting library components for users to get dashboard related information at the event and to use the information in their own application.

Event types

- The start action of adding a library component
- The end action of adding a library component
- The start action of deleting a library component
- The end action of deleting a library component

Basic Information

- Catalog information
- Dashboard information
- Library component information

For detailed usages about the Dashboard Listener API, see JReport Javadoc com.jinfonet.web.modules.dashboard package in `<install_root>\help\api`. Also, there is a sample implementation file DashboardListenerDemo.java in `<install_root>\help\samples\APIDashboardListener`.

Applying the implementations of the Dashboard Listener API

Different uses should have different implementations of the Dashboard Listener API. The implementations can be managed on the server by administrators. After a new implementation is imported to the server and enabled, it will be able to take effect immediately without restarting the server.

Besides, after an implementation is uploaded to the server, you can apply it in the URL which is used for running dashboards. The URL way has higher priority than the enabled implementation on the JReport Administration page.

To import and enable an implementation on the JReport Administration page:

1. Log onto the JReport Administration page.
2. Click **Configuration** on the system toolbar and then select **Dashboard Listener** from the drop-down menu.
3. In the Dashboard Listener panel, click **New Dashboard Listener**.
4. In the New Dashboard Listener dialog, click the **Browse** button to locate the implementation file which could be a zip or a jar file. Then click **Next**.
5. From the Class Name drop-down list, choose a class name which is defined in the implementation file.
6. Click + beside the Target box, then in the Select Target dialog, specify the target dashboards you would like the implementation to apply to. Click **OK** to confirm the selection.
7. In the Description text field, enter a description for the implementation if necessary.
8. Select the **Enabled** checkbox to make the implementation enabled.
9. Click **Finish** and the registered class will take effect right away.

See also Dashboard Listener panel for additional information.

To apply an implementation via URL:

To apply the dashboard listener via URL, append the key-value pair "jrd_dsh_listener=PackageName.ClassName" in the URL which is used for running dashboards.

For example, if you want your implemented class TestListener to take effect on a new created dashboard (not the target defined on the JReport Administration page), you can define it in the URL. Such URL will not affect the Dashboard Listener Table on the JReport Administration page.

```
http://localhost:8888/dashboard/app/entry/run.jsp?jrs.authorization=YWRtaW46YWRtaW4%
3D&jrd_dsh_listener=com.jinfonet.TestListener&jrd_resext={"active":0,"reslst":
[{"name":"/USERFOLDERPATH/admin/Dashboard1.dsh","ver":"-1"}]}
```

Dynamic connection API

JReport Server manages dynamic connections by the DynamicConnectionManager. Users can call jet.server.api.admin.AdminService.getDynamicConnectionManager() to get the DynamicConnectionManager object, and call its methods addDynamicConnection(...), updateDynamicConnection(...) or removeDynamicConnection(...) to maintain dynamic connections for multiple tenancy clients by programs.

When running a report, the server calls the DynamicConnectionProvider (the built-in provider calls DynamicConnectionManager internally) to get dynamic connections for all used data sources at first and then passes the dynamic connection properties into the engine. The engine will merge the changed properties with the original catalog connection properties to setup the database connection.

Users can implement jet.server.api.dynamiccon.DynamicConnectionProvider to get dynamic connections from their own systems and then register a customized DynamicConnectionProvider into JReport Server by adding the option server.custom.DynamicConnectionProvider in the server.properties file in `<install_root>\bin`.

For detailed usages about the dynamic connection API, see JReport Javadoc jet.server.api.dynamiccon package in `<install_root>\help\api`.

Related topics:

- Configuring dynamic connections

Dynamic security API

JReport Server provides API for dynamic security management.

- **getDynamicSecurities**
  Gets all dynamic security objects.
- **addDynamicSecurity**
  Adds a new dynamic security object.
- **deleteDynamicSecurity**
  Deletes an existing dynamic security object.
- **getCatalogPath**
  Gets the full path of the catalog resource.
- **getSecurityFileName**
  Gets the name of the security file.
- **getSecurityFile**
  Gets the security file for downloading.
- **setSecurityFile**
  Sets the security file after uploading.
- **setDefault**
  Sets the value of the property IsDefault to true/false.
- **APIConst.TAG_SECURITY_FILE_NAME**
  Defines the security at runtime.

Related topics:

- Managing dynamic security

Information Bus API

Information Bus API enables you to access information container and get or put information in the container.

- **InformationBus**
  It is used to transmit information in JReport Server. It contains three kinds of information containers, global level, organization level and user level, and you can get or put information in the containers.

- **InformationBusManager**
  It is used to get the Information Bus instance from the Information Bus Manager.

- **InformationContainer**
  It is used to store user information of the following types:
  - **LONG_TIME**
    The information exists in the container until it is removed or the container it belongs to is removed.
  - **SPECIFIED_TIME**
    The information exists in the container until the time you specify arrives or you remove it, or the container it belongs to is removed.
  - **ONCE_TIME**
    The information will be removed from the container once you get or remove it, or the container it belongs to is removed.

- **InfoLifeCycleType**
  It is used to specify the life cycle type of the information, which can be LONG_TIME, SPECIFIED_TIME OR ONCE_TIME.

Reference: See the JReport Javadoc in `<install_root>\help\api` for detailed usages of Information Bus API.

API demos

Here is list of the JReport Server demo programs.

- Programs in `<install_root>\help\samples\APICluster`:
  - **DemoLoadBalancer.java**
    Demo to implement the jet.server.api.cluster.LoadBalancer for randomly selecting a clustered server. For details, see Balancing the server load.
  - **DemoRemoteDispatcher.java**
    An example of dispatching request to remote clustered server according to Round-Robin algorithm. For details, see Dispatching RMI Server Pages requests in multiple server environment.
- Programs in `<install_root>\help\samples\APIParameter`:
  - **DemoParameterGenerator.java**
    An example of implementing the jet.server.api.custom.schedule.ParameterGenerator for setting dynamic parameter values. For details, see Setting dynamic parameter values when running/ scheduling report via API.
- Programs in `<install_root>\help\samples\APISecurity`:
  - **CustomizedSendFileAuthorizor.java**
    Demonstrates implementation of SendFileAuthorizor, which is used to control users' accessibility to certain files in server machine when they access the server via http.
  - **DemoAuthenticationProvider.java**
    Demo for implementation of the jet.server.api.custom.security.AuthenticationProvider interface. For details, see Customized implementation of the Security API.
  - **DemoAuthorizationProvider.java**
    Demo for implementation of the jet.server.api.custom.security.AuthorizationProvider interface. For details, see Customized implementation of the Security API.
- Programs in `<install_root>\help\samples\APISecurity\AddPrincipal`:
  - **AddGroup.java**
    An example of creating groups and getting the current information of group.
  - **AddRole.java**
    An example of creating roles and getting the current information of role.
  - **AddUser.java**
    An example of creating users and getting the current information of user.
  - **SecurityGroupCover.java**
    An example of creating groups and configuring other security settings.
  - **SecurityRoleCover.java**
    An example of creating roles and configuring other security settings.
  - **SecurityUserCover.java**
    An example of creating users and configuring other security settings.
- Programs in `<install_root>\help\samples\APISecurity\LoginLogout`:
  This folder holds JSP source files that experiment with how the two versions of checkLogin() work. The files provide a way to access pages protected by checkLogin() by doing various forms of login to see how the process works. After doing a login, the user can do a logout, in order to establish the same initial state to try another scenario.

  The experiments relate to accessing files that are protected by a call to one of the checkLogin() mehtods which validates that a user is logged into the session or logs a user into the session based on the current HTTP Request.

  Comments in the entry point page, loginIndex.jsp, describe how the files work and how to deploy them for use.
  - **loginIndex.jsp**
    Main entry point for the group of JSP files. This is the dispatcher and home page for the group.
  - **loginDoCheckLogin.jsp**
    This does the work to run the right JSP page that matches the scenario chosen by the user in loginIndex.jsp.
  - **loginUtilCheckLogin.jsp**
    This is the page that is protected by a call to HttpUtil.checkLogin(). After checkLogin() allows it to run, it reports on how it was run and what happened to allow it.
  - **loginSessionCheckLogin.jsp**
    This is the page that is protected by a call to HttpUserSessionManager.checkLogin(). After checkLogin() allows it to run, it reports on how it was run and what happened to allow it.
  - **loginTunOnSSO.jsp**
    This is a helper page to register that the ExternalAuthorized package com.example.MyExternalAuthorized is to be used by JReport Server. This turns on the Single Sign On functionality. This page is only useful (and is only given as a choice by loginIndex.jsp) when the com.exmple.MyExternalAuthorized package is deployed and findable on the class path. Read the comments in the source to the package found in the SingleSignOn folder next to this one.
  - **loginTunOffSSO.jsp**
    This is a helper page to unset the ExternalAuthorized that is currently set in the server. This turns off the Single Sign On functionality.
  - **loginLogout.jsp**
    This is a helper page that logs out a user from the session in order to have a standard starting state.
- Programs in `<install_root>\help\samples\APISecurity\SingleSignOn`:
  This folder holds sample source files that show examples of implementing a Single Sign On system to allow an existing application and JReport Server to co-operate with logging in users with the same web server.

  There are two example implementations of the HttpExternalAuthorized java interface. These implementations provide the Single Sign On functionality. Each file has comments inside that give directions on how to install and use them.
  - **CustomHttpExternalAuthorized.java**
    Implementation of HttpExternalAuthorized to load when JReport Server starts up.
  - **com\example\MyExternalAuthorized.java**
    Implementation of HttpExternalAuthorized written as a package to load on demand while JReport Server is running.

  One file is the source for a servlet that can be used to dynamically set which instance of an HttpExternalAuthorized is used by JReport Server. This would be compiled and deployed using local technique for deploying a servlet.
  - **CustomServlet.java**
    Demo to set implementation of HttpExternalAuthorized into the report server.

  The rest of the files in the folder are JSP files that can be placed into the web server to run through several scenarios that show how an application and JReport Server JSP pages can co-operate to manage user logins within a Single Sign On system. Each file has comments inside that explain what it does and how it fits into the demonstration. These JSP files can be copied to the folder `public_html\jinfonet` where they can be run with a standard install of JReport Server.
  - **customIndex.jsp**
    Entry point JSP page for the demo, to run through the various scenarios that show how Single Sign On behaves. This page includes operations for logging out, to recreate the state of a user not being logged in. It also has controls for turning on and off the Single Sign On functionality.
  - **customViewReport.jsp**
    This is the requested report used in the demo to show how Single Sign On supports authentication for it. It runs a simple report which is displayed in the browser.
  - **customNoAuth.jsp**
    Runs the scenario where nothing special is done in the run-report request to authenticate the user.
  - **customParamAuth.jsp**
    Runs the scenario where the User ID is passed in via a query parameter in the run-report request as a way to authenticate the user.
  - **customSessionAuth.jsp**
    Runs the scenario where the User ID is put into the servlet session variable emulating how the application login would do this, as a way to authenticate the user.
  - **customTurnOnSSO.jsp**
    Helper page that sets the instance of the HttpExternalAuthroized class registered with JReport Server, to turn on the Single Sign On functionality.
  - **customTurnOffSSO.jsp**
    Helper page that unsets the instance of the HttpExternalAuthroized class registered with JReport Server, to turn off the Single Sign On functionality.
- Programs in `<install_root>\help\samples\APIServer`:
  - **APIDemoDeployRpt.java**
    An example of publishing catalogs, reports, or folders to the server.
  - **APIDemoDynamicExportTask.java**
    An example of scheduling a customized task. For details, see Scheduling a customized task using User Task.
  - **APIDemoPublishRpt.java**
    An example of publishing a scheduled task.
  - **APIDemoReportEngine.java**
    An example of using the ReportEngine to run a report, export/print a report, and set the where portion.
  - **APIDemoRunAndExportReport.java**
    An example of exporting a report to some formats, and then saving them to local disk.
  - **APIDemoRunReport.java**
    An example of running a report.
  - **APIDemoRunReportWithTimeout.java**
    An example of running reports with timeout.
  - **APIDemoSendEMail.java**
    An example of publishing a report to e-mail.
  - **DemoTrigger.java**
    This demo creates a trigger, submit a schedule bound with the trigger, then fire the trigger to run the schedule.
  - **ViewHtmlPipelineServlet.java**
    This is a servlet demo of viewing report as HTML pipeline format.
- Programs in `<install_root>\help\samples\APITaskListener`:
  - **TestTaskListener.java**
    An example to demonstrate how to implement users' own scripts before and after task running and how to get the information about a task. For details, see Adding TaskListener.
- Programs in `<install_root>\help\samples\JSPSamples`:
  - **ApplyUserCSS**
    An example of applying user defined CSS to the HTML report result. For details, see Applying a user defined CSS to HTML result file.
  - **DHTMLRunAndExport**
    An example of using JSP to build URL to run a report in JReport Desktop Viewer Bean. After copying the contents of the folder (demo) to where JSP can be run, for example, `<install_root>\public_html`, you can run the demo using the URL `http://localhost:8888/demo/main.jsp`.
  - **OpenMultipleReports**
    Demo JSPs for opening multiple reports in one session. For details, see Opening multiple reports in one session.
  - **PrintReport**
    These JSP files show how to print reports by using ViewerApplet. For details, see Using JSPs to print page reports.
  - **RunReport**
    Demo JSPs for runing report in Advanced mode to various formats. To run the demo, copy the two JSP files to `<intall_root>\public_html\jinfonet`, and then access getRptDescPage2.jsp (runReport2.jsp is called by getRptDescPage2.jsp) using the URL `http://localhost:8888/jinfonet/getRptDescPage2.jsp`.
  - **JinfonetGourmetJavaDemo**
    Demo JSPs used by the tutorial manual. For details, refer to Track 6: Integrating Reports with JSP in the *JReport Tutorial*.
- Programs in `<install_root>\help\samples\URLSamples`:
  - **ExampleCodeForTags.html**
    The sample code for different tags. For details about tags, see Appendix 8: Page Report Studio tag library.
  - **TestURL.html**
    Some example URLs.

RMI demos

Here are some demo RMI programs that JReport provides. They are available in `<install_root>\help\samples\APIRemoteServer`.

- **RemoteAPIDemoFileService.java**
  A demo about using remote report server.
- **RemoteAPIDemoPublishRpt.java**
  A demo about publishing reports. You can submit a schedule to export a report to version, disk, or e-mail.
- **RemoteAPIDemoRunAndExportReport.java**
  A demo about running reports and exporting report result to other formats.
- **RemoteAPIDemoRunReportWithTimeout.java**
  A demo about the Large Report Manager. This demo runs a report and marks it as a large report by setting the timeout value, and then returns the large report information.

Before running these demos, remember to do the following:

1. Modify the server.properties file in the `<install_root>\bin` directory to set the property server.rmiserver.enable to **true**.
2. Add **-Djrs.rmi.auth_file="%authFileName%"** to the command line that starts the required program.
3. You need to set the classpath environment variable. Append the following jar files to the class path that compile and run applications which call the Server API:

   ```
   <install_root>\lib\JRESServlets.jar;
   <install_root>\lib\JREngine.jar;
   <install_root>\lib\servlet.jar;
   <install_root>\lib\log4j-1.2.8.jar;
   <install_root>\lib\sac.jar;
   <install_root>\lib\commons-codec-1.2.jar
   ```

JReport Catalog API

JReport Catalog API is intended for those who want to work with a catalog programmatically. You can create a catalog and manage the objects in it with the methods provided by the Catalog API instead of using the GUI design mode in JReport Designer. Compared with the Catalog Bean, the Catalog API offers greater functionality. As with a catalog in JReport Designer, the Catalog API provides methods which enable you to:

- Create a connection.
- Add and delete objects such as tables, queries, formulas, parameters, and WHERE portions.
- Modify a query and update UDS.
- Get and modify object information.

The combination of the Catalog API and the Design API provides you with a Java development environment that enables you to create any report to meet your requirements.

Pick a task from the following for details about how to use the Catalog API:

- Installing the Catalog API
- Programming with the Catalog API

Installing the Catalog API

When you install JReport Designer, which by default is located in `C:\JReport\Designer`, the Catalog API is also installed at the same time. After installation, you will have the following components in `<install_root>\lib`:

- report.jar
- JREngine.jar
- log4j-1.2.8.jar (Jar file used for JReport Engine logging system)
- sac.jar (Jar file used for installing Catalog API with Design API product)
- ...

The Catalog API class is stored in the archive file report.jar. You will need to do the two steps below before you can compile and run the program:

1. Set the class path environment variable. Append the following string to your class path that starts the Catalog API:

   ```
   <install_root>\lib\report.jar;<install_root>\lib\sac.jar;<install_root>\lib
   \JREngine.jar;<install_root>\lib\log4j-1.2.8.jar
   ```

2. In your Java program you must set the licensed user and Design API license key.

   ```
   DesignerUserInfo userInfo=new DesignerUserInfo("UID","Design API Key");
   ```

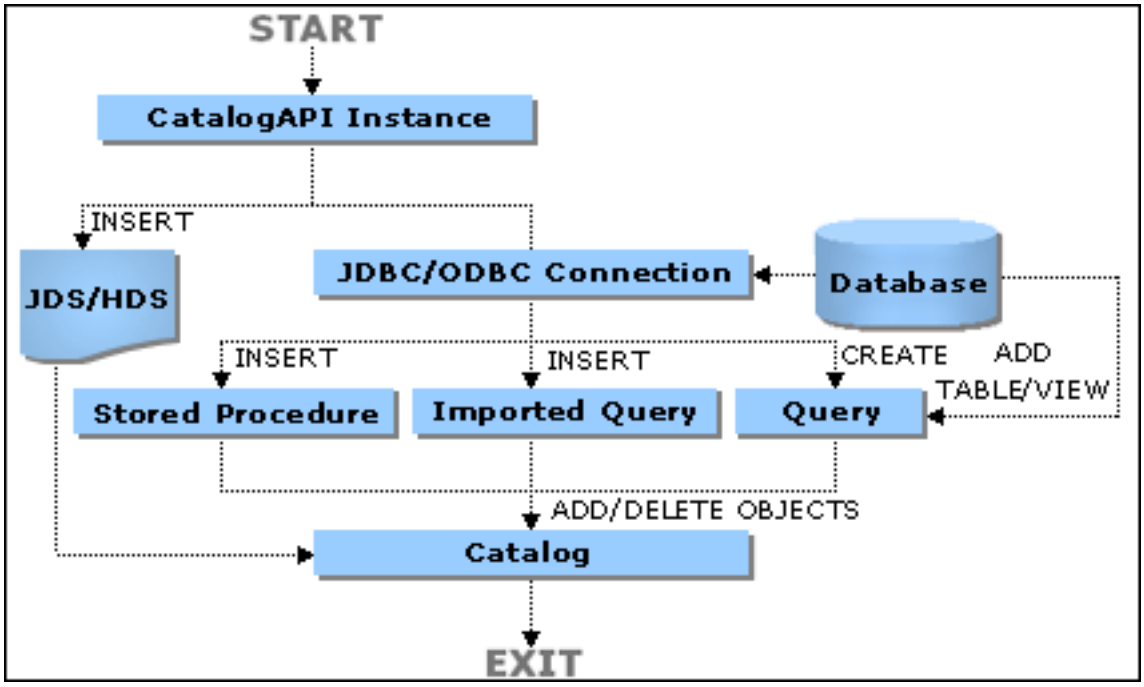
3. Set the reporthome parameter equal to the root path of JReport Designer. You can set this in your environment or with the -Dreporthome argument to your JVM.

JReport Server also includes the Catalog API. If you prefer to use the libraries from JReport Server include the jar file <server_install_root>\lib\JRESServlets.jar in your class path instead of report.jar. You would then need to use the Server Design API license key rather than the Design API license key.

Programming with the Catalog API

JReport Catalog API is the principal class that manipulates and accesses catalogs. The application creates a Catalog API instance, and then uses the Catalog API methods to operate the catalog. For more details about summaries of FIELD, CONSTRUCTOR and METHOD, see the JReport Javadoc jet.api package in `<install_root>\help\api`, which includes all documents generated by Javadoc. Also, you are provided with some sample files in `<install_root>\help\samples\APICatalog`.

With the methods in JReport Catalog API, you can do the following:



Select a task from the following for details:

- Preparations
- Using the Catalog API
- An illustrative example

Preparations

Before you can use the Catalog API to perform certain tasks, you will have to create a Designer object and then get a Catalog API instance.

Creating a Designer object

To create a Designer object, use the constructor Designer(String path, String name, DesignerUserInfo user) in the Design API. The constructor has three parameters: the catalog path, catalog name, and the user ID and license key provided by Jinfonet.

The path needs to be a valid path to an existiing directory. The catalog name can be the name of an existing catalog when you want to open a catalog, or the name of a new catalog when you want to create a catalog. If you want to create a new catalog, the path directory should not already contain a catalog file.

To create the DesignerUserInfo instance, use the following constructor providing the user ID and the Server Designer License Key or Designer License Key you received when you purchased JReport.

```
DesignerUserInfo userInfo=new DesignerUserInfo(Uid, key);
```

Getting a Catalog API instance

To get a Catalog API instance, use the method getCatalogAPI() in the Design API. You need to first get an instance from Designer as follows.

```
Designer desg = new Designer(catalogPath, catalogName, userInfo);
CatalogAPI catalog = desg.getCatalogAPI();
```

Using the Catalog API

After creating a Designer object and getting a Catalog API instance, you can then use the Catalog API to perform tasks as required. For example, you can use it to set up a connection in a catalog, insert an object into the catalog, and modify the query of the catalog.

The following sections describe how to program with the Catalog API:

- Creating a connection
- Inserting an object
- Inserting a table/view
- Inserting a stored procedure
- Importing an SQL file
- Inserting a UDS
- Inserting a query
- Inserting a business cube, report cube or business view
- Inserting a formula, summary, parameter or a WHERE portion
- Deleting an object
- Modifying a query
- Getting objects in a catalog
- Refreshing the reference table of a catalog
- Saving a catalog

Creating a connection

The following methods are used to insert a database connection and return the handle of the newly inserted connection. If not successful, a null value is returned:

- insert(boolean bUseJDBCODBCBridge, String name, String desc, String url, String driver)
- insert(boolean bUseJDBCODBCBridge, String name, String desc, String url, String driver, ConnectionInfo info)
- insert(String dataSourceName, boolean bUseJDBCODBCBridge, String name, String desc, String url, String driver)
- insert(String dataSourceName, boolean bUseJDBCODBCBridge, String name, String desc, String url, String driver, ConnectionInfo info)

Parameters

- UseJDBCODBCBridge - Indicates to connect to the database with a JDBC-ODBC Bridge.
- name - The name of the connection.
- desc - The description for the connection.
- url - The URL used to connect to the database.
- driver - The database driver.
- info - The connection information.

Note: The class ConnectionInfo is used to create the connection information. For details, refer to the JReport Javadoc jet.api package in `<install_root>\help\api`, which includes all documents generated by Javadoc.

Inserting an object

The following methods are used to insert an object into a catalog and to return the handle of the newly inserted object:

- insert(String name, int type)
- insert(String dataSourceName, String name, int type)

Parameters

- name - The object name.
- type - The object type. For more information, refer to the JReport Javadoc jet.api.CatalogAPI class in `<install_root>\help\api`, which includes all documents generated by Javadoc.

Inserting a table/view

The following methods are used to insert a table, view or synonyms into a catalog and to return the handle of the newly inserted object:

- insert(String catalogName, String schemaPattern, String tableName, int type)
- insert(String catalogName, String schemaPattern, String tablePattern, int type, boolean setMappingName)

Parameters

- catalogName - The catalog name.
- schemaPattern - The schema pattern.
- tablePattern - The table name.
- type - The table type.

Inserting a stored procedure

The following methods are used to insert a stored procedure into a connection and to return the handle of the newly inserted procedure.

- insert(String procName,String catalog,String schema,String name,String remarks,int iType)
- insert(String procName, String catalog, String schema, String name, String remarks, int iType)

Parameters

- procName - The procedure mapping name.
- catalog - The catalog name of the procedure.
- schema - The schema name of the procedure.
- name - The procedure name.
- remarks - The remarks on the procedure.
- iType - The result type of the procedure.

Importing an SQL file

The following methods are used to import an SQL file and to return the handle of the newly inserted SQL file:

- insert(String SQLName, String filename)
- insertSql(String dataSourceName, String SQLName, String filename)

Parameters

- SQLName - The SQL name.
- filename - The imported file name.

Inserting a UDS

The following methods are used to insert a user defined data source and to return the handle of the newly inserted UDS:

- insert(String strDSName, String strClassName, String strParameter, UDSColumnInfo udsColInfo)
- insert(String dataSourceName, String strDSName, String strClassName, String strParameter, UDSColumnInfo udsColInfo)

Parameters

- strDSName - The UDS name.
- strClassName - The name of the UDS class.
- strParameter - The parameter value.
- udsColInfo - The specified column information.

Inserting a query

The following methods are used to insert a query into a catalog and to return the handle of the newly inserted query:

- insert(String qryName, QueryFieldInfo queryFieldInfo,QueryJoinInfo queryJoinInfo, QueryQBEInfo queryQBEInfo, QueryAndInfo queryAndInfo)
- insert(String dataSourceName, String qryName, QueryFieldInfo queryFieldInfo, QueryJoinInfo queryJoinInfo, QueryQBEInfo queryQBEInfo, QueryAndInfo queryAndInfo)

Parameters

- qryName - The query name.
- queryFieldInfo - The table and fields information.
- queryJoinInfo - The join information.
- queryQBEInfo - The QBE information.
- queryAndInfo - The where condition information.

Reference: For details about the classes QueryFieldInfo, QueryJoinInfo, QueryQBEInfo and QueryAndInfo, refer to the JReport Javadoc jet.api package in `<install_root>\help\api`, which includes all the documents generated by Javadoc.

Inserting a business cube, report cube or business view

The following methods are used to insert a business cube or report cube into a catalog and to return the handle of the newly inserted object if it is inserted successfully, otherwise null if there is a failure:

- insertBusinessView(String dataSourceName, String queriableName, String businessViewName)
- insertBusinessView(String dataSourceName, String queriableName, String businessViewName, boolean isLogicView=true)

To insert a business view, you can only use the following method:

- insertBusinessView(String dataSourceName, String queriableName, String businessViewName, boolean isLogicView=false)

Parameters

- dataSourceName - The mapping name of the data source in the catalog to which you want to insert the object.
- queriableName - The queriable object's mapping name. The queriable object in JReport catalog could be a query, UDS, stored procedure or imported SQL. If queriableName is a valid name of an existing queriable object, a report cube or business view will be created; if it is null or does not exist, a business cube will be created.
- businessViewName - The name of the business cube, report cube or business view.
- isLogicView - Indicates whether to insert a business cube/report cube or a business view. If it is true, a business cube/report cube will be created, otherwise, a business view will be created.

Adding elements to business cube/report cube •Deprecated•

When you have a business cube or report cube created successfully, you can add elements into it. A business cube/report cube can contain the following elements: categories, dimension objects, measure objects, and detail information objects.

The following methods are used to insert a cube element into a business cube or report cube and to return the handle of the newly inserted object if it is inserted successfully, otherwise null if there is a failure:

- insertBLCategory(String parentHandle, BLCategoryInfo info)
- insertBLDimension(String parentHandle, BLDimensionInfo info)
- insertBLMeasure(String parentHandle, BLMeasureInfo info)
- insertBLDetail(String parentHandle, BLDetailInfo info)

Parameters

- parentHandle - The handle of the parent object. It can be a business cube/report cube or a category.
- info - The definition class of the cube element. It can contain its children's definition.

Adding elements to business view

Similarly, you can add elements into a business view using the following methods, which include categories, group objects, aggregation objects and detail objects:

- insertBusinessViewCategory(String parentHandle, BusinessViewCategoryInfo info)
- insertBusinessViewGroup(String parentHandle, BVGroupInfo info)
- insertBusinessViewAggregation(String parentHandle, BVAggregationInfo info)
- insertBusinessViewDetail(String parentHandle, BVDetailInfo info)

Parameters

- parentHandle - The handle of the parent object. It can be a business view or a category.
- info - The definition class of the view element. It can contain its children's definition.

See an example: The sample TestCatalogBC.java in `<install_root>\help\samples\APICatalog` shows how to create a report cube using the Catalog API.

Inserting a formula, summary, parameter or a WHERE portion

The following methods are used to insert a formula, summary, parameter or a WHERE portion into a catalog and to return the handle of the newly inserted object:

- insert(String formulaName, String desc, String expression)
- insert(String summaryName, String desc, int functionType, String fieldName, String groupByFld)
- insert(String parameterName, String desc, String prompt, String type, String default Value)
- insert(String wherePortionName, String desc, WherePortionInfo whereportionInfo)

Parameters

- formulaName - The name of the formula.
- desc - The description of the object.
- expression - The expression.
- summaryName - The name of the summary.
- functionType - The type of the function.
- fieldName - The field name calculated on.
- groupByFld - The field name grouped by.
- parameterName - The parameter Name.
- prompt - The input text prompt.
- type - The parameter data type.
- defaultValue - The defaultValue of the parameter.
- wherePortionName - The WHERE portion name.
- whereportionInfo - The WHERE portion information.

Deleting an object

The method delete(Stringhandle) is used to delete an object from its parent node and to return true if the specified object is removed.

Parameter

- handle - The handle of the object that is to be deleted.

Modifying a query

You can perform the following manipulations when modifying a query.

Adding or deleting a table

- **set(String dataSourceName, String qryName, String tablename, String columnname, boolean isFormula)**
  Adds a new table to a query and returns a Boolean value: true if successful, otherwise false.
- **deleteQueryTable(String dataSourceName, String qryName, String tablename)**
  Deletes the selected table in a query and returns a Boolean value: true if successful, otherwise false.

Note: The tables that are to be added should exist in the specified catalog.

Parameters

- qryName - The query name.
- tablename - The name of the table that is to be inserted or deleted.
- selectAllFields - Specifies whether or not to insert a table with all its fields.

Adding or deleting a field

- **set(String dataSourceName, String qryName, String tablename, String columnname, boolean isFormula)**
  Adds a new field to a query and returns a Boolean value: true if successful, otherwise false.
- **deleteQueryField(String dataSourceName, String qryName, String tablename, String columnname)**
  Deletes the selected fields in a query and returns a Boolean value: true if successful, otherwise false.

Parameters

- qryName - The query name.
- tablename - The table name of the inserted or deleted field.
- isFormula - Indicates whether the inserted field is a formula.
- columnname - The column name of the deleted field.

Adding or deleting a join

- **set(String dataSourceName, String qryName, String tableFrom, String columnFrom, String operator, String tableTo, String columnTo, boolean isSQL92, int outerJoin)**
  Adds a new join to a query and returns a Boolean value: true if successful, otherwise false.
- **deleteQueryJoin(String dataSourceName, String qryName, String tableFrom, String columnFrom, String operator, String tableTo, String columnTo)**
  Deletes joins in a query and returns a Boolean value: true if successful, otherwise false.

Parameters

- qryName - The query name.
- tableFrom - The name of the table from which the join links.
- columFrom - The name of the column from which the join links.
- tableTo - The name of the table to which the join links.
- columnTo - The name of the column to which the join links.
- operator - Operator of the join to be inserted or deleted.

Adding or deleting a QBE condition

- **setQBE(String dataSourceName, String qryName, String tablename, String columnname, String expression)**
  Adds a new QBE condition to a query and returns a Boolean value: true if successful, otherwise false.
- **deleteQBE(String dataSourceName, String qryName, String tablename, String columnname, String expression)**
  Deletes the QBE condition in a query and returns a Boolean value: true if successful, otherwise false.

Parameters

- qryName - The query name.
- tablename - The name of the table where the QBE condition will be added or deleted.
- columnname - The name of the column where the QBE condition will be added or deleted.
- expression - The expression of the QBE condition.

Adding or Deleting a where condition

- **setCondition(String dataSourceName, String qryName, String sExpression1, String sOperator, String sExpression2, String sLogic)**
  Adds a new where condition to a query and returns a Boolean value: true if successful, otherwise false.
- **deleteCondition(String dataSourceName, String qryName, String sExpression1, String sOperator, String sExpression2, String sLogic)**
  Deletes the where conditions in a query and returns a Boolean value: true if successful, otherwise false.

Parameters

- qryName - The query handle.
- sLogic - The logic string of the where condition that is to be inserted or deleted.
- sExpression1 - The first expression of the where condition that is to be inserted or deleted.
- sOperator - The operator of the where condition that is to be inserted or deleted.
- sExpression2 - The second expression of the where condition that is to be inserted or deleted.

Getting objects in a catalog

Getting tables

- **getTables(String dataSourceName)**
  Gets tables, views and synonyms in a connection and returns the mapping name array.
- **getTables(boolean bIncludeSystemTable)**
  Gets tables, views and synonyms in a data source and returns the name array.

Getting stored procedures

- **getProcedureNames(String dataSourceName)**
  Gets all the stored procedures in a catalog and to return the mapping name array.

Getting SQLs

- **getSQLs(String dataSourceName)**
  Gets all the SQL files in a catalog and to return the mapping name array.

Getting UDSs

- **getUDSs(String dataSourceName)**
  Gets all the UDSs in a catalog and to return the mapping name array.

Getting queries

- **getQueries(String dataSourceName)**
  Gets queries in a catalog and returns the mapping name array.
- **getNewQueryName()**
  Gets a default name for a new query, which must be unique. The name for the first query in a catalog is query1. The following query is query2, then query3 and so on.

Getting columns

- **getColumns(String dataSourceName, String queryName, boolean allColumns)**
  Gets columns in a query and returns the table name and column real names.
- **getColumnMappingnames(String dataSourceName, String queryName)**
  Gets column mapping names in a query and returns the table name and column mapping names.
- **getColumnsCanBeGroupedBy(String dataSourceName, String queryName)**
  Gets the columns which can be grouped by in a query.

Parameters

- queryName - The query name.
- allColumns - Selects all columns in the specified tables.

Getting fields

- **getDBFields(String dataSourceName, String qryName)**
  Gets DBFields that can be used in a query and returns the mapping name array.
- **getMappingnames(String dataSourceName, String DSName)**
  Gets the field mapping names in a data source and returns the table name and column mapping names.

Parameters

- qryName - The query name.
- DSName - The data source object name, such as Query, SQL, UDS, and Procedure.

Getting formulas, summaries or parameters

- **getFormulae(String dataSourceName)**
  Gets formulae that can be used in a catalog and returns the mapping name array.
- **getFormulaeCanBeGroupedBy(String dataSourceName, String queryName, String groupBy)**
  Gets formulae that are valid to a query and returns the formula names.
- **getSummaries(String dataSourceName)**
  Gets summaries that can be used in a catalog and returns the mapping name array.
- **getSummaryCanBeSortedBy(String dataSourceName, String queryName, String groupBy)**
  Gets summaries that can be sorted by.
- **getParameter(String dataSourceName)**
  Gets parameters in a catalog and returns the mapping name array.
- **getParameterCanBeGroupedBy(String dataSourceName)**
  Gets parameters that can be grouped by.

Parameters

- queryName - The query name.
- dataSourceName - The datasoure name.
- groupBy - The groupBy field.

Getting joins

- **getJoins(String dataSourceName, String queryName)**
  Gets joins in a query and returns the join info array.

  Note: String[i][0] = Table "From" Name, String[i][1] = Column "From" Name, String[i][2] = Table "To" Name, String[i][3] = Column "To" Name

Parameter

- queryName - The query name.

Getting AND or QBE condition

- **getAndConditions(String dataSourceName, String queryName)**
  Gets AND conditions in a query and returns a condition info array.

  Note: String[i][0] = "Logic", String[i][1] = "Expression1" String[i][2] = "Operator", String[i][3] = "Expression2"
- **getQBEInfo(String dataSourceName, String queryName)**
  Gets the QBE condition in a query and returns a QBE info array.

  Note: String[i][0] = column Name, String[i][j](j>=1) = condition expression of this column.

Parameter

- queryName - The query name.

Getting report names

- **getNewReportName()**
  Gets a default name for a new report, which must be unique. The name for the first report in a catalog is report1, after which the following reports will be named report2, report3 and so on.

•

Refreshing the reference table of a catalog

You can use the method public boolean refreshReference(CrossReferenceConfigure referenceConfigure) to refresh the reference table of a catalog.

Parameter

- CrossReferenceConfigure referenceConfigure - The configuration information that defines which data sources in a catalog and which types of object in the data sources will be added to the reference table.

Saving a catalog

The following methods are used to save catalogs:

- **save()**
  Saves an open catalog and returns a Boolean value: false if it fails, otherwise true.
- **saveAs(String path, String name)**
  Saves an open catalog as a new catalog file. You can save the catalog as a binary catalog using the extension .cat, or an XML format catalog using the extension .cat.xml.

An illustrative example

You are provided with a sample program that demonstrates how to use the Catalog API to create or modify a catalog - TestCatalogAPI.java that is located in `<install_root>\help\samples\APICatalog`.

Here, TestCatalogAPI.java has been used as an example to explain how the methods work in the Catalog API.

- **Creating a Designer object**

```
DesignerUserInfo userInfo=new DesignerUserInfo("UID",
"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX");
desg = new Designer(path, catName, userInfo);
```

  If the catalog named catName (MyReports.cat) exists, it will be opened. Otherwise it will be created.

- **Getting a CatalogAPI instance**

```
CatalogAPI cat = dr.getCatalogAPI();
```

- **Creating a connection**

```
ConnectionInfo info = new ConnectionInfo("sa", "");
String con = cat.insert("", false,"Jinfonet demo", "demo db",
"jdbc:hsqldb:C:\\JReport\\Designer\\Demo\\db\\JRDemo", "org.hsqldb.jdbcDriver", info);
```

- **Inserting objects into a catalog**

  For example, insert a table named Customers:

```
tb.addTable("", null, "PUBLIC", "CUSTOMERS", "Customers");
```

- **Inserting a formula named "total"**

```
String formulahandles=cat.insert("total",cat.FORMULA);
FormulaInfo formulainfo=new FormulaInfo("total","total formula","@\"Unit Price\" *
@Quantity - @Discount");
cat.set(formulahandles,formulainfo);
```

- **Importing an SQL file**

```
cat.insert("SQL1","sqlfile.txt");
```

- **Saving**

```
cat.save();
```

- **Compiling and running the demo program**

  To compile and run the demo program, you should add the required classes to the class path.

  After running the program, a new catalog file MyReports.cat will be created in `C:\JReport\Designer\Demo\MyReports`. Be sure to create this empty directory first.

```
C:\JReport\Designer\help\samples\APICatalog>javac -classpath "C:\JReport\Designer\lib
\report.jar;C:\JReport\Designer\lib\sac.jar;C:\JReport\Designer\lib\JREngine.jar;C:
\JReport\Designer\lib\log4j-1.2.8.jar;C:\test\classes" TestCatalogAPI.java
```

```
C:\JReport\Designer\help\samples\APICatalog>java -classpath "C:\JReport\Designer\lib
\jrengine.jar;C:\JReport\Designer\lib\sac.jar;C:\JReport\Designer\lib\report.jar;C:
\JReport\Designer\lib\log4j-1.2.8.jar;C:\JReport\Designer\help;C:\test\classes" -
Dreporthome=C:\JReport\Designer TestCatalogAPI -path=C:\JReport\Designer\Demo
\MyReports -catalog=MyReports.cat -log=C:\JReport\Designer\logs\designer.log
```

•

JReport Design API

Besides editing a report with the design panel in JReport Designer, you can also edit a report with the Design API. By creating a Designer instance in your Java program, you can modify a report with the methods provided by the Design API. The Design API is for designing a report with your own application.

JReport Design API fully realizes designing a report programmatically. That is, with the Design APIs you can edit the layout of a report all in code. You can change the property values, or modify the report by inserting objects such as DBFields, formulas, text objects, and charts. You can also create and design a report with the Design API.

The following sections explain the Design API in detail:

- Design API packages & license
- Installing the Design API
- Getting started
- Programming with the Design API

Note: The Design API only supports Page Reports (.cls) and does not support Web Reports (.wls) and Library Components (.lc).

•

Design API packages & license

There are two JReport Design API packages: the workstation version and the server version. The workstation package is the Design API, and the server package is the Server Design API. The main differences between these two packages are:

- There are some special methods used for editing catalogs, queries, reports, and parameters. These methods are included in the Design API and Server Design API. When invoking these methods in the Server Design API, you also have to provide an additional parameter: UID (used to distinguish between different users, and can be a Number or a String). However, when invoking them in the Design API, you do not need to provide this parameter.
- The Design API can only edit one report at a time, while the Server Design API allows concurrent users to modify their reports.
- The Multi-User Design API included with Server Design API only supports V7 and earlier style report templates and not multi-tab page reports supported by Design API.

JReport Design API and JReport Server Design API use an independent license from other JReport products. You need to contact Jinfonet sales to obtain this special license key if you want to use either JReport Design API or JReport Server Design API on your machine with JReport Designer or JReport Server. The license will be applied to your program instead of being applied to the installation of JReport Design API or JReport Server Design API.

The keys for JReport Design API and JReport Server Design API are different. For example, you cannot use the JReport Design API key on JReport Server Design API, and vice versa. Which key is required is based on where you set your class path at runtime. If the classes are obtained from Server\lib you need to use the Server Design API key. If the classes are obtained from Designer\lib you need to use the Design API key.

To use the key you need to create a DesignerUserInfo instance. To create the DesignerUserInfo instance, use the following constructor providing the user ID and the Server Designer License Key or Designer License Key you received when you purchased JReport.

```
DesignerUserInfo userInfo=new DesignerUserInfo(Uid, key);
Designer desg = new Designer(catalogPath, catalogName, userInfo);
```

If you want to use JReport Design API or JReport Server Design API package for workstations without using JReport Designer and JReport Server, you will need to obtain a valid license for each work station.

•

Installing the Design API

The full Design API packages are included in JReport Designer and JReport Server. JReport Designer includes a single threaded Design API, while JReport Server includes a multi-threaded Design API. JReport Design API and JReport Server Design API packages can also be downloaded and installed to your local machine. After which, you can then program with them without having to install any other JReport products.

Installing the Design API by JReport Designer

When you install JReport Designer, the Design API will be installed at the same time. After installation, you will have the components listed below in `<designer_install_root>\lib`:

- JREngine.jar
- report.jar
- sac.jar
- commons-codec-1.2.jar
- log4j-1.2.8.jar (Jar file used for the JReport Engine logging system)

Installing the Design API by JReport Server

When you install JReport Server, the Server Design API will be installed at the same time. After installation, you will have the components listed below in `<server_install_root>\lib`:

- JREngine.jar
- JREServlets.jar
- sac.jar
- commons-codec-1.2.jar
- log4j-1.2.8.jar (Jar file used for the JReport Engine logging system)

Installing independent JReport Design API

Ask the Jinfonet Support (support@jinfonet.com) for the independent Design API installation file. Then, run the file and follow the prompts to install.

After you have installed JReport Design API by running the installation file, you will need to take the following two steps before you compile and run the program:

1. Set the classpath environment variable. Then, append the following string to your class path that starts the Design API:

   ```
   <designerAPI_install_root>\lib\designAPI.jar;<designerAPI_install_root>\lib\log4j-1.2.8.jar;<designerAPI_install_root>\lib\sac.jar;<designerAPI_install_root>\lib\commons-codec-1.2.jar;<designerAPI_install_root>\lib\hsqldb.jar
   ```

2. Set the -Dreporthome parameter equal to the root path of the Design API.

Installing independent JReport Server Design API

Ask the Jinfonet Support (support@jinfonet.com) for the independent Server Design API installation file. Then, run the file and follow the prompts to install. We don't recommend using these libraries, we recommend you use the Design API libraries included in JReport Server.

After you have installed JReport Server Design API by running the installation file, you will need to take the following two steps before you compile and run the program:

1. Set the classpath environment variable. Then, append the following string to your class path that starts the Server Design API:

   ```
   <svrdesignerAPI_install_root>\lib\serverDesignAPI.jar;<svrdesignerAPI_install_root>\lib\log4j-1.2.8.jar;<svrdesignerAPI_install_root>\lib\sac.jar;<svrdesignerAPI_install_root>\commons-codec-1.2.jar;<svrdesignerAPI_install_root>\hsqldb.jar
   ```

2. Set the -Dreporthome parameter to be the same as the root path of the Server Design API.

Note: The Design API classes are stored in the archive files: report.jar and JREngine.jar. You will need to take the following two steps before you can compile and run the program:

1. Set the classpath environment variable. Append the following string to your class path that starts the Design API:

   ```
   <designer_install_root>\lib\report.jar;<designer_install_root>\lib\JREngine.jar;<designer_install_root>\lib\log4j-1.2.8.jar;<designer_install_root>\lib\sac.jar;<designer_install_root>\commons-codec-1.2.jar;
   ```

2. Add the jar files for your DBMS JDBC drivers.
3. Set the -Dreporthome parameter to be the same as the root path of JReport Designer.
4. You will need to register the Design API licensed user and license key in your source code.

Note: The Server Design API classes are stored in the archive files: JRESServlets.jar and JREngine.jar. You will need to take the following two steps before you can compile and run the program:

1. Set the classpath environment variable. Append the following string to your class path that starts the Design API:

   ```
   <server_install_root>\lib\JRESServlets.jar;<server_install_root>\lib\JREngine.jar;<server_install_root>\lib\log4j-1.2.8.jar;<server_install_root>\lib\sac.jar;<server_install_root>\commons-codec-1.2.jar;
   ```

2. Add the jar files for our DBMS JDBC drivers.
3. Set the -Dreporthome parameter to be the same as the root path of JReport Server.
4. You will need to register the Server Design API licensed user and license key in your source code.

•

Getting started

To help you start using the Design API, here is a simple example which shows how to use the JReport Design API package to compile and run a Java class that returns the connection information from a catalog file.

1. Compile the sample code TellMeConnection.java in `<install_root>\help\samples\APICatalog`.

   ```
   javac -classpath <install_root>\lib\JREngine.jar;<install_root>\lib\sac.jar;
   <install_root>\lib\report.jar TellMeConnection.java
   ```

2. Set the license key for Design API using the method `dr.setUserInfo("UID","XXXXXXXXXXX")` in TellMeConnection.java.

3. Run the sample code.

   ```
   java -classpath ".;<install_root>\lib\JREngine.jar;<install_root>\lib\report.jar;
   <install_root>\lib\sac.jar;<install_root>\lib\log4j-1.2.8.jar" -
   Dreporthome=<install_root> TellMeConnection –path=<catalog path> -catalog=<catalog
   name>
   ```

   For example,

   ```
   java -classpath "c:\JReport\Designer\lib\JREngine.jar;c:\JReport\Designer\lib
   \report.jar;c:\JReport\Designer\lib\sac.jar;c:\JReport\Designer\lib\log4j-1.2.8.
   jar;C:\JReport\Designer\lib\hsqldb.jar" -Dreporthome=D:\JReport\Designer
   TellMeConnection -path=C:\JReport\Designer\Demo\Reports\TutorialReports -
   catalog=JinfonetGourmetJava.cat
   ```

   And the output should be as follows:

   Name: Jinfonet demo
   Driver: org.hsqldb.jdbcDriver
   URL: `jdbc:hsqldb:c:\JReport\Designer\Demo\db\JRDemo`
   User: sa
   Password
   TimestampFormat yyyy-MM-dd HH:mm:ss.SSS
   TimeFormat HH:mm:ss
   DateFormat yyyy-MM-dd
   ExtraNamePattern Default(JDBC)

•

Programming with the Design API

The class API is the root of the class hierarchy of JReport Design API (Design API and Catalog API). It is an abstract class and provides a series of editing methods for users. The Design API class has an API as a super class.

Designer and MultiUserDesigner are the principal classes that manipulate reports. The application creates a Designer instance and calls Designer methods to create or modify a report. For details of summary of FIELD, CONSTRUCTOR and METHOD, see JReport Javadoc jet.api package in `<install_root>\help\api`, which includes all documents generated by Javadoc.

The following topics explain how to use the Design API:

- Preparations
- Using the Design API
- Illustrative examples

Preparations

Before you can use the Design API to perform certain tasks, you need to first make the following preparations.

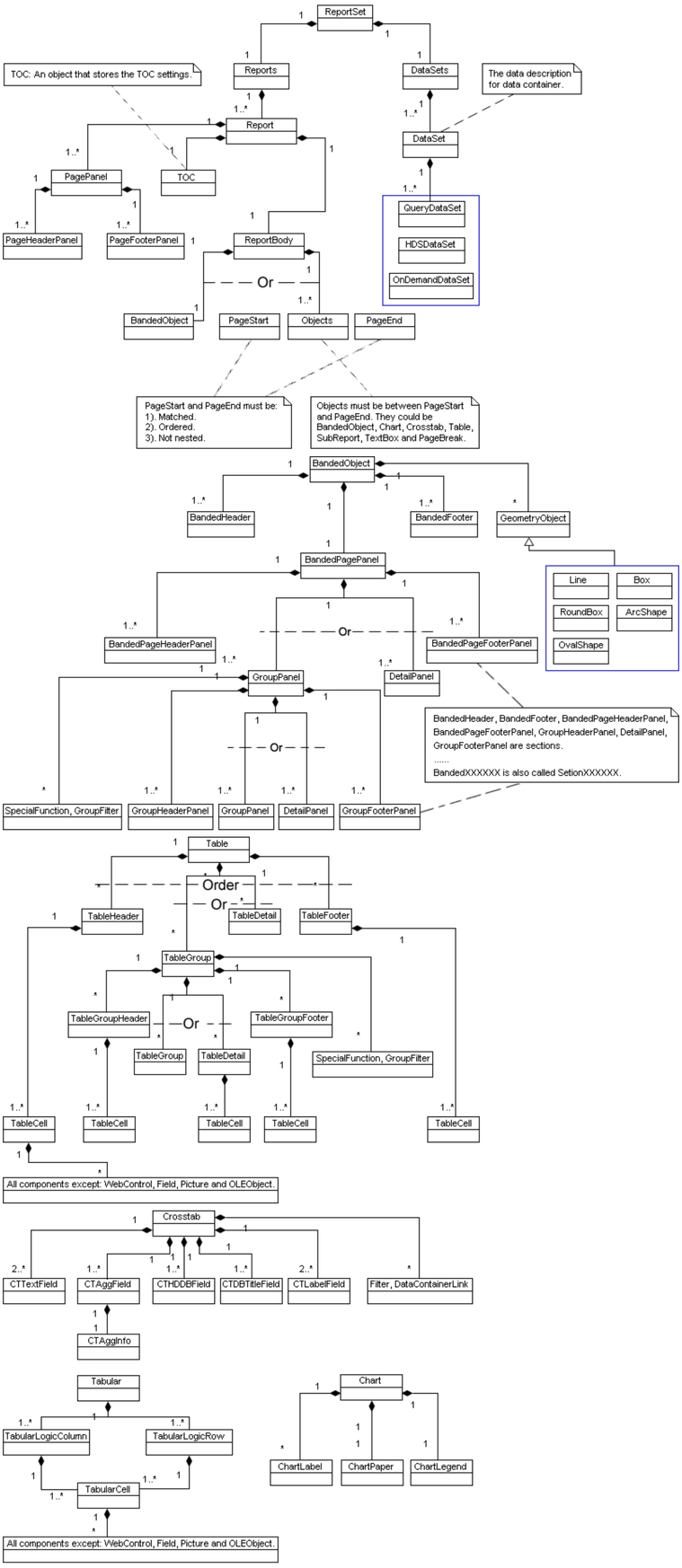
Setting the license key

The license information is set via a method called setUserInfo(String uid, String key), where UID is the user ID, and key is the license key. This method should be called before you can call any other Design API methods. The UID is the license user ID you received with the purchase of JReport Server and JReport Designer. The license key will be either Server Designer License Key or Designer License Key depending on your class path setting to Server\lib or Designer\lib respectively.

```
DesignerUserInfo userInfo=new DesignerUserInfo(Uid, key);
Designer desg = new Designer(catalogPath, catalogName, userInfo);
```

Getting handle information

All the objects in a report are organized into and browsed from a tree structure. For each report, the Report Inspector consists of a report objects tree at the top of the view panel and its corresponding reporting features or properties at the bottom. Selecting any object node on the tree also selects the same element in the report design area. The following diagrams cover all objects that can be used in a report with their relationships.



Every node and object in the above trees is identified by a HANDLE in JReport Design API. Therefore, before you can create or edit a report with the Design API, you will first have to get the handle information. Super class API provides the following methods to get handles of different nodes and different types.

- **getHandles()**
  Gets all the handles of a report and returns a handle array.
- **getHandles(String handle)**
  Gets handles of the sub node in the current node and returns a handle array. If it fails, it will return null.
- **getHandles(String handle, int type)**
  Gets sub node handles of the same types for the given node and returns a handle array. If it fails, it will return null.
- **getHandles(String handle, int type, int depth)**
  Gets sub node handles of the same types for the given node and returns a handle array. The depth is described below. If it fails, it will return null.
- **getParent(String handle)**
  Gets the parent handle of an object and returns the parent handle. If it fails, it will return null.

Parameters

- handle - Parent node handle when getting sub node handles. Object handle when getting parent handle.
- type - Class type value defined in the API class.
- depth - This is related to the current level in the report tree. When the depth is -1, all handles in the report will be returned. When the depth is 0, the handles of the current level will be returned. When the depth is number n, the handles of n levels and the current level will be returned.

Using the Design API

After making the required preparations, you can now use the Design API to perform tasks.

The following topics detail how to program with the Design API to achieve certain goal:

- Creating and deleting a report
- Opening and closing a report
- Inserting and deleting an object
- Getting property information
- Getting report metadata information
- Getting a Catalog API instance
- Getting objects and object information
- Getting the instance name and class type of an object
- Moving objects
- Getting editing information
- Exiting from the editing status
- Concurrent control

•

Creating and deleting a report

- **create(String name, String query)**
  Creates a new report with a specified name, and returns the handle of the new report. If it fails, a null value will be returned.
- **createReportSet(String name)**
  Creates a new report with a specified name.
- **deleteReport(java.lang.String name)**
  Deletes an existing report with the specified name, and returns true if the report has been successfully removed.

Parameters

- name - The specified report name that is to be created or deleted.
- query - The query name on which the report is created.

Notes:

- The report should be created based on the existing catalog file, otherwise you will get an error.
- The query should exist in the given catalog, otherwise you will get an error.

•

Opening and closing a report

To open or close a report, you can use the following methods:

- **open(String name)**
  Opens an existing report and returns the handle of the report.
- **close(String handle)**
  Saves any changes and closes the open report tab in the current report.
- **closeReportSet()**
  Saves any changes and closes the current report.
- **quit(String handle)**
  Closes an open report without saving any changes.

Parameters

- name - Name of the report that is to be opened.
- handle - Ancestor handle of the open report.

•

Inserting and deleting an object

To insert or delete an object into a report, you can use the methods and parameters below:

- **insert(String parent, int type, String name)**
  Inserts an object into the parent node.
- **insert(String parent, int type, String name, String mapping)**
  Inserts a DBField, parameter, formula or summary into the parent node.
- **insert(String parent, String name, CTRowColFieldInfo colInfo[], CTRowColFieldInfo rowInfo[], CTAggFieldInfo aggInfo[])**
  Insert a crosstab object and its children into the parent node.
- **insert(String parent, CTRowColFieldInfo colInfo, CTRowColFieldInfo rowInfo, CTAggFieldInfo aggInfo)**
  Inserts crosstab children into the crosstab.
- **insert(String parent, String name, String paperName, int charttype, String gName1, String gName2, String value, ChartLegendInfo chartLegendInfo, ChartLabelInfo chartLabelInfo)**
  Inserts a chart object and its children into the parent node.
- **insert (String parent, int type, String name, String topSection , String bottomSection)**
  Inserts a shape object into the report and returns the handle of the new shape.
- **insert(String parent, String name, GroupInfo groupInfo)**
  Inserts a group/sort into the parent node.
- **delete(String handle)**
  Deletes an object from its parent node.

Parameters

- parent - Parent handle node.
- type - Class type that you want to insert.
- name - Instance name of the new object.
- mapping - Mapping name of the database field, parameter, formula or summary.
- colInfo - Field information array of the Column.
- rowInfo - Field information array of the Row.
- aggInfo - Field information array of the Aggregate.
- paperName - Instance name of ChartCoordinatepaper object.
- charttype - Type of Chart.
- group1 - Mapping name of the first group.
- group2- Mapping name of the second group. It can be null.
- value - Mapping name of a summary.
- chartLegendInfo - Field information array of the chart legend.
- chartLabelInfo - Field information array of the chart title and notes.
- handle - Handle of the object that is to be deleted.
- topSection - Handle of the top section attached by the shape.
- bottomSection - Handle of the bottom section attached by the shape.
- groupInfo - The group information.

G-Z

Querying geometry information

The following functions are used to query the geometry values of different types and return the corresponding values.

- gmlsf.getlength(geometry, String, number)
- gmlsf.getarea(geometry, String, number)
- gmlsf.getcoordinatedimension(geometry, String, number)
- gmlsf.getdimension(geometry, String, number)
- gmlsf.getisclosed(geometry, String, number)
- gmlsf.getisvalid(geometry, String, number)
- gmlsf.getisvalidreason(geometry, String, number)
- gmlsf.getisvalidreason(geometry, String, number, number)
- gmlsf.getisvalidreason(geometry, String, number, number, number)

Note: The gmlsf.getisvalidreason(geometry, String, number) function is used to get a long fields text of validity reason for a geometry.

- gmlsf.getnumelements(geometry, String, number)
- gmlsf.getnumpoints(geometry, String, number)
- gmlsf.getpointn(geometry, String, number, number)
- gmlsf.getsrid(geometry, String, number)
- gmlsf.getstartpoint(geometry, String, number)
- gmlsf.getendpoint(geometry, String, number)
- gmlsf.getexteriorring(geometry, String, number)
- gmlsf.getinteriorringn(geometry, String, number, number)
- gmlsf.getnuminteriorring(geometry, String, number)

Note: The gmlsf.getnuminteriorring(geometry, String, number) function is used to get the number of interior rings in a polygon.

- gmlsf.getnumgeometries(geometry, String, number)
- gmlsf.getgeometryn(geometry, String, number, number)
- gmlsf.getcentroid(geometry, String, number)
- gmlsf.getpointonsurface(geometry, String, number)

Note: The gmlsf.getpointonsurface(geometry, String, number) function is used to get a point that is guaranteed to be on the surface.

- gmlsf.getdistance(geometry, geometry, number)
- gmlsf.getdistance(geometry, geometry, number, number)

Note: The gmlsf.getdistance(geometry, geometry, number, number) function is used to get a 2-dimensional distance between two geometries in the units specified.

- gmlsf.getdistance(geometry, geometry, number, number, number)

Note: The gmlsf.getdistance(geometry, geometry, number, number, number) function is used to get the distance between two geometries in the units specified, in the dimension specified.

- gmlsf.getdistance(geometry, geometry, number, number, number, number)

Note: The gmlsf.getdistance(geometry, geometry, number, number, number, number) function is used to get the distance between two geometries in the units specified, in the dimension specified, and with the specified accuracy.

- gmlsf.getdimensionbounds(geometry, String, number, number)
- gmlsf.getdimensionbounds(geometry, String, number, number, number)

Note: The gmlsf.getdimensionbounds(geometry, String, number, number, number) function is used to get the bounds of a specified dimension in the units specified, in the dimension specified, and with the specified accuracy.

- gmlsf.getbounds(geometry, String, number)
- gmlsf.getbounds(geometry, String, number, number)

Note: The gmlsf.getbounds(geometry, String, number, number) function is used to get the bounds in the units specified, in the dimension specified.

- gmlsf.getbounds(geometry, String, number, number, number)

Note: The gmlsf.getbounds(geometry, String, number, number, number) function is used to get the bounds in the units specified, in the dimension specified, and with the specified accuracy.

- gmlsf.getbounds(geometry, String, number, number, number, number)

Note: The gmlsf.getbounds(geometry, String, number, number, number, number) function is used to get the bounds in the units specified, in the dimension specified, and with the specified accuracy, and in the specified coordinate system.

- gmlsf.getbounds(geometry, String, number, number, number, number, number)

Note: The gmlsf.getbounds(geometry, String, number, number, number, number, number) function is used to get the bounds in the units specified, in the dimension specified, and with the specified accuracy, and in the specified coordinate system, and in the specified output dimension.

- gmlsf.getbounds(geometry, String, number, number, number, number, number, number)

Note: The gmlsf.getbounds(geometry, String, number, number, number, number, number, number) function is used to get the bounds in the units specified, in the dimension specified, and with the specified accuracy, and in the specified coordinate system, and in the specified output dimension, and with the specified output precision.

- gmlsf.getbounds(geometry, String, number, number, number, number, number, number, number)

Note: The gmlsf.getbounds(geometry, String, number, number, number, number, number, number, number) function is used to get the bounds in the units specified, in the dimension specified, and with the specified accuracy, and in the specified coordinate system, and in the specified output dimension, and with the specified output precision, and with the specified output flags.

- gmlsf.getenvelope(geometry, String, number)
- gmlsf.getenvelope(geometry, String, number, number)
- gmlsf.getenvelope(geometry, String, number, number, number)
- gmlsf.getenvelope(geometry, String, number, number, number, number)
- gmlsf.getenvelope(geometry, String, number, number, number, number, number)
- gmlsf.getenvelope(geometry, String, number, number, number, number, number, number)
- gmlsf.getenvelope(geometry, String, number, number, number, number, number, number, number)

Note: The gmlsf.getenvelope(geometry, String, number, number, number, number, number, number, number) function is used to get the envelope in the units specified, in the dimension specified, and with the specified accuracy, and in the specified coordinate system, and in the specified output dimension, and with the specified output precision, and with the specified output flags, and with the specified output srid.

- gmlsf.getcoordinate(geometry, String, number, number)
- gmlsf.getcoordinate(geometry, String, number, number, number)
- gmlsf.getcoordinate(geometry, String, number, number, number, number)
- gmlsf.getcoordinate(geometry, String, number, number, number, number, number)
- gmlsf.getcoordinate(geometry, String, number, number, number, number, number, number)
- gmlsf.getcoordinate(geometry, String, number, number, number, number, number, number, number)

Note: The gmlsf.getcoordinate(geometry, String, number, number, number, number, number, number, number) function is used to get the coordinate in the units specified, in the dimension specified, and with the specified accuracy, and in the specified coordinate system, and in the specified output dimension, and with the specified output precision, and with the specified output flags, and with the specified output srid.

- gmlsf.getcoordinate(geometry, String, number, number, number, number, number, number, number, number)

Note: The gmlsf.getcoordinate(geometry, String, number, number, number, number, number, number, number, number) function is used to get the coordinate in the units specified, in the dimension specified, and with the specified accuracy, and in the specified coordinate system, and in the specified output dimension, and with the specified output precision, and with the specified output flags, and with the specified output srid, and with the specified output coordinate system.

Querying geometry information

The following functions are used to get the geometry values of different types. They will return NULL if it is not the type of geometry.

- gmlsf.getcurvelength(geometry, String, number, number)
- gmlsf.getcurvearea(geometry, String, number, number)
- gmlsf.getcurvecoordinates(geometry, String, number, number)
- gmlsf.getcurvedimension(geometry, String, number, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number, number, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number, number, number, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number, number, number, number, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number, number, number, number, number, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number, number, number, number, number, number, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number, number, number, number, number, number, number, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number, number, number, number, number, number, number, number, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number, number, number, number, number, number, number, number, number, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number, number, number, number, number, number, number, number, number, number, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number, number, number, number, number, number, number, number, number, number, number, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number, number, number, number, number, number, number, number, number, number, number, number, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number, number, number, number, number, number, number, number, number, number, number, number, number, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number, number, number, number, number, number, number, number, number, number, number, number, number, number, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number, number, number, number, number, number, number, number, number, number, number, number, number, number, number, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number, number, number, number, number, number, number, number, number, number, number, number, number, number, number, number, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number, number, number, number, number, number, number, number, number, number, number, number, number, number, number, number, number, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number, number, number, number, number, number, number, number, number, number, number, number, number, number, number, number, number, number, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number)
- gmlsf.getcurvedimensionbounds(geometry, String, number

	Kindergartenzone	Kindergarten is a school or school-like institution for children aged 3 to 6 years. Kindergarten is a school or school-like institution for children aged 3 to 6 years. Data type: Boolean
	Kinder	Kinder are children. Data type: Boolean
	Kinderbetreuungsangebot	Kinderbetreuungsangebot is a place where children are cared for, for example, in a kindergarten, nursery school, or day care center. Data type: Boolean
	Kinderbetreuungsangebotzone	Kinderbetreuungsangebotzone is a place where children are cared for, for example, in a kindergarten, nursery school, or day care center. Data type: Boolean
	Kinderbetreuung	Kinderbetreuung is the care of children by someone other than their parents or guardians, typically in a kindergarten, nursery school, or day care center. Data type: Boolean
	Kinderbetreuungszone	Kinderbetreuungszone is the care of children by someone other than their parents or guardians, typically in a kindergarten, nursery school, or day care center. Data type: Boolean
	Kinderbetreuungsstätte	Kinderbetreuungsstätte is a place where children are cared for, for example, in a kindergarten, nursery school, or day care center. Data type: Boolean
	Kinderbetreuungsstättezone	Kinderbetreuungsstättezone is a place where children are cared for, for example, in a kindergarten, nursery school, or day care center. Data type: Boolean
	Kinderbetreuungszentrum	Kinderbetreuungszentrum is a place where children are cared for, for example, in a kindergarten, nursery school, or day care center. Data type: Boolean
	Kinderbetreuungszentrumzone	Kinderbetreuungszentrumzone is a place where children are cared for, for example, in a kindergarten, nursery school, or day care center. Data type: Boolean
	Kindergarten	Kindergarten is a school or school-like institution for children aged 3 to 6 years. Data type: Boolean
	Kindergartenzone	Kindergartenzone is a school or school-like institution for children aged 3 to 6 years. Data type: Boolean
	Kindergeld	Kindergeld is a monthly payment to parents or guardians of children under 18 years of age, regardless of their income or employment status. Data type: Boolean
	Kindergeldzone	Kindergeldzone is a monthly payment to parents or guardians of children under 18 years of age, regardless of their income or employment status. Data type: Boolean
	Kindergrenze	Kindergrenze is the maximum age at which children are considered children for tax purposes. Data type: Boolean
	Kindergrenzezone	Kindergrenzezone

Ancestor		Ancestor
Ancestor		Ancestor
Ancestor		Ancestor

Rain-Gauge		Gauge for measuring the quantity of rain falling in a given time. Syn. Pluviometer. Ant. Anemometer. Adj. Rain-gauge.
Rain-gauge		Adj. Rain-gauge. Gauge for measuring the quantity of rain falling in a given time. Syn. Pluviometer. Ant. Anemometer.
Rain-gauging		Noun. Verbal. Measuring the quantity of rain falling in a given time.
Rainbow		Noun. Concrete. Arch of many colours, formed by the refraction of the sun's rays in a shower. Adj. Rainbow.
Rainbow-coloured		Adj. Rainbow. Arch of many colours, formed by the refraction of the sun's rays in a shower.
Rainbow-tinted		Adj. Rainbow. Arch of many colours, formed by the refraction of the sun's rays in a shower.
Raindrop		Noun. Concrete. Drop of water, formed by the condensation of the vapour in the atmosphere. Syn. Rain-drop.
Rainfall		Noun. Concrete. Quantity of water falling in a given time. Syn. Rain-fall

	WordWrap	Specifies whether to wrap the text to the width. Data type: Boolean
Subreport	AnchorDispValue	Specifies a string or formula to display for the TOC entry for this object in the TOC Browser. The TOCAnchor property must be set to true for this property to take effect. Data type: String
	Background	Specifies the background color of the object. Data type: String
	BorderColor	Specifies the color of the border of the object. Data type: String
	BorderWidth	Specifies the width of the border in inches or centimeters. Data type: Float
	BottomLine	Specifies the line style of the bottom border of the object. Data type: Enumeration
	Cache	Specifies whether to cache the dataset for this object in the data buffer so other objects which use the same dataset can share the data rather than perform their own SQL query. Data type: Boolean
	class	Specifies a CSS class to be applied to the object which is a valid class in the CSS file. Data type: String
	ColumnIndex	Specifies the X coordinate of the object relative to its parent container when exported to Excel or CSV, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Integer
	Embedded	If false, the page header and footer of the subreport will not be shown. Data type: Boolean
	ExportToCSV	Specifies whether to include the object when exporting the report to CSV. Data type: Boolean
	ExportToDHTML	Specifies whether to include the object when exporting the report to Page Report Result. Data type: Boolean
	ExportToXLS	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
	ExternalCSSClassValue	Specifies a class selector to be applied to the object when exported as HTML. Data type: String
	ExternalID	It is mapped to the HTML attribute *id*. This attribute specifies a name to the object, which must be unique in the report. Data type: String
	ExternalStyle	It is mapped to the HTML attribute *style*. This attribute specifies style information for the object. Data type: String
	Height	Specifies the height of the object, in inches or centimeters. Data type: Float
	Invisible	Specifies whether to show the object in the design area and in the report result. All formulas and calculations will still be performed if the property is set to true. Data type: Boolean
	Lang	It is mapped to the HTML attribute *lang*. This attribute specifies the base language of the object's attribute values and text content. Data type: String
	LeftLine	Specifies the line style of the left border of the object. Data type: Enumeration
	MaxPageNum	Specifies the maximum number of pages in the data buffer. Data type: Integer
	OnNewSheet	Specifies whether to put the subreport in a new sheet when exporting the report to Excel. Data type: Boolean
	position	Specifies the position of the object. • **static** - The object is positioned at the location in which it is inserted. The X, Y and other position-related properties will be hidden or disabled. The position of the object is affected by preceding objects in the same container. • **absolute** - The object will be located at the position specified by dragging and dropping or by setting its X and Y property values. The position of the object is not affected by preceding objects in the same container. Data type: String **Note:** This property only affects objects with a parent container that is the report body, a tabular cell, or a text box.
	Postfix	Specifies how to add postfix number to the sheets in which the subreport will be put when exporting the report to Excel. • **Add postfix number to every page** - Specifies to add the postfix number to every sheet for the subreport (Subreport1, Subreport2, Subreport3, Subreport4, ...). • **Add postfix number only to additional pages** - Specifies to add the postfix number to the subreport sheets starting from the second one (Subreport, Subreport1, Subreport2, Subreport3, ...). It is meaningful to specify the property value when the OnNewSheet property is set to true. Data type: String
	RecPerPage	Specifies the number of records in each page in the data buffer. Data type: Integer
	RecordLocation	Specifies the calculation point for the properties which use formulas. • **Default** - The values of the properties are calculated in the default location where they are located. • **Page Header** - The values of the properties are calculated in the page header. • **Page Footer** - The values of the properties are calculated in the page footer. Data type: String
	RightLine	Specifies the line style of the right border of the object. Data type: Enumeration
	RowIndex	Specifies the Y coordinate of the object relative to its parent container when exported to Excel or CSV, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Integer
	SheetName	Specifies the name of the new sheet in which the subreport will be put. Make sure the name specified here isn't used by any other sheets. It is meaningful to specify the property value when the OnNewSheet property is set to true. Data type: String **Note:** The characters "|", ":", "/", "?", "\", "*", "]", "[" cannot be displayed in the sheet name in Excel and the last character in the sheet name cannot be single quotation mark ('). If any one of them is used when you specify the property, it will be replaced by "_" in Excel.
	SRQueryName	Displays the name of the query used by the subreport. This property is read only.
	SRSecurity	Displays the security name (record level security) used by the subreport. This property is read only.
	SRSourceName	Displays the name of the data source used by the subreport. If nothing is displayed, the default data source will be used. This property is read only.
	Style	The property can be used in two ways. • Specifies a style to be applied to the object. This works when a StyleGroup has been specified at the report sheet level and when there are styles in the StyleGroup that can be applied to the object. • Specifies a CSS selector to be applied to the object. Data type: String
	Suppressed	Specifies whether to show the object in the design area and in the report result. All formulas and calculations will be skipped if the property is set to true. Data type: Boolean **Note:** When both the Invisible and Suppressed properties of an object are set to true, Suppressed has the higher priority.
	SuppressedIfNoRecords	Specifies whether to display the object in the report result when no record is returned to its parent data container. Data type: Boolean
	SuppressEmpty	Specifies whether to display the object in the report result when no record is returned to it. Data type: Boolean
	TOCAnchor	Specifies whether to include the object in the TOC Browser for the report. Data type: Boolean
	TopLine	Specifies the line style of the top border of the object. Data type: Enumeration
	Width	Specifies the width of the object, in inches or centimeters. Data type: Float
	X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property is ignored if the position property is set to static. Data type: Float
	Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property is ignored if the position property is set to static. Data type: Float
Line	BottomAttachCol	The four Excel properties, TopAttachCol, TopAttachRow, BottomAttachCol and BottomAttachRow, work together to control the position of a drawing object in the exported Excel file. BottomAttachCol specifies the X coordinate of the lower right corner of the drawing object in the exported Excel file, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Float
	BottomAttachPosX	Specifies the horizontal coordinate of the bottom right point of the line in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	BottomAttachPosY	Specifies the vertical coordinate of the bottom right point of the line in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	BottomAttachRow	The four Excel properties, TopAttachCol, TopAttachRow, BottomAttachCol and BottomAttachRow, work together to control the position of a drawing object in the exported Excel file. BottomAttachRow specifies the Y coordinate of the lower right corner of the drawing object in the exported Excel file, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Float
	class	Specifies a CSS class to be applied to the object which is a valid class in the CSS file. Data type: String
	ExportToDHTML	Specifies whether to include the object when exporting the report to Page Report Result. Data type: Boolean
	ExportToXLS	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
	ExternalCSSClassValue	Specifies a class selector to be applied to the object when exported as HTML. Data type: String
	ExternalID	It is mapped to the HTML attribute *id*. This attribute specifies a name to the object, which must be unique in the report. Data type: String
	ExternalStyle	It is mapped to the HTML attribute *style*. This attribute specifies style information for the object. Data type: String
	ExternalTitle	It is mapped to the HTML attribute *title*. This attribute offers advisory information about the object. Data type: String
	Invisible	Specifies whether to show the object in the design area and in the report result. All formulas and calculations will still be performed. Data type: Boolean
	Lang	It is mapped to the HTML attribute *lang*. This attribute specifies the base language of the object's attribute values and text content. Data type: String
	LineColor	Specifies the color of the line. Data type: String
	LineStyle	Specifies the style of the line. Data type: Enumeration
	LineWidth	Specifies the thickness of the line. Data type: Float
	RecordLocation	Specifies the calculation point for the properties which use formulas. • **Default** - The values of the properties are calculated in the default location where they are located. • **Page Header** - The values of the properties are calculated in the page header. • **Page Footer** - The values of the properties are calculated in the page footer. Data type: String
	Style	The property can be used in two ways. • Specifies a style to be applied to the object. This works when a StyleGroup has been specified at the report sheet level and when there are styles in the StyleGroup that can be applied to the object. • Specifies a CSS selector to be applied to the object. Data type: String
	SuppressedIfNoRecords	Specifies whether to display the object in the report result when no record is returned to its parent data container. Data type: Boolean
	TopAttachCol	The four Excel properties,TopAttachCol, TopAttachRow, BottomAttachCol and BottomAttachRow, work together to control the position of a drawing object in the exported Excel file. TopAttachCol specifies the X coordinate of the upper left corner of the drawing object in the exported Excel file, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Float
	TopAttachPosX	Specifies the horizontal coordinate of the top left point of the line in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	TopAttachPosY	Specifies the vertical coordinate of the top left point of the line in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	TopAttachRow	The four Excel properties, TopAttachCol, TopAttachRow, BottomAttachCol and BottomAttachRow, work together to control the position of a drawing object in the exported Excel file. TopAttachRow specifies the Y coordinate of the upper left corner of the drawing object in the exported Excel file, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Float
Box	Background	Specifies the background color of the object. Data type: String
	BorderColor	Specifies the color of the border of the object. Data type: String
	BorderStyle	Specifies the line style of the box border. Data type: Enumeration
	BorderWidth	Specifies the width of the object's border, in inches or centimeters. Data type: Float
	BottomAttachCol	The four Excel properties, TopAttachCol, TopAttachRow, BottomAttachCol and BottomAttachRow, work together to control the position of a drawing object in the exported Excel file. BottomAttachCol specifies the X coordinate of the lower right corner of the drawing object in the exported Excel file, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Float
	BottomAttachPosX	Specifies the horizontal coordinate of the bottom right point of the box in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	BottomAttachPosY	Specifies the vertical coordinate of the bottom right point of the box in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	BottomAttachRow	The four Excel properties, TopAttachCol, TopAttachRow, BottomAttachCol and BottomAttachRow, work together to control the position of a drawing object in the exported Excel file. BottomAttachRow specifies the Y coordinate of the lower right corner of the drawing object in the exported Excel file, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Float
	class	Specifies a CSS class to be applied to the object which is a valid class in the CSS file. Data type: String
	ExportToDHTML	Specifies whether to include the object when exporting the report to Page Report Result. Data type: Boolean
	ExportToXLS	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
	ExternalCSSClassValue	Specifies a class selector to be applied to the object when exported as HTML. Data type: String
	ExternalID	It is mapped to the HTML attribute *id*. This attribute specifies a name to the object, which must be unique in the report. Data type: String
	ExternalStyle	It is mapped to the HTML attribute *style*. This attribute specifies style information for the object. Data type: String
	ExternalTitle	It is mapped to the HTML attribute *title*. This attribute offers advisory information about the object. Data type: String
	Invisible	Specifies whether to show the object in the design area and in the report result. All formulas and calculations will still be performed. Data type: Boolean
	Lang	It is mapped to the HTML attribute *lang*. This attribute specifies the base language of the object's attribute values and text content. Data type: String
	RecordLocation	Specifies the calculation point for the properties which use formulas. • **Default** - The values of the properties are calculated in the default location where they are located. • **Page Header** - The values of the properties are calculated in the page header. • **Page Footer** - The values of the properties are calculated in the page footer. Data type: String
	Style	The property can be used in two ways. • Specifies a style to be applied to the object. This works when a StyleGroup has been specified at the report sheet level and when there are styles in the StyleGroup that can be applied to the object. • Specifies a CSS selector to be applied to the object. Data type: String
	SuppressedIfNoRecords	Specifies whether to display the object in the report result when no record is returned to its parent data container. Data type: Boolean
	TopAttachCol	The four Excel properties, TopAttachCol, TopAttachRow, BottomAttachCol and BottomAttachRow, work together to control the position of a drawing object in the exported Excel file. TopAttachCol specifies the X coordinate of the upper left corner of the drawing object in the exported Excel file, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Float
	TopAttachPosX	Specifies the horizontal coordinate of the top left point of the box in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	TopAttachPosY	Specifies the vertical coordinate of the top left point of the box in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	TopAttachRow	The four Excel properties, TopAttachCol, TopAttachRow, BottomAttachCol and BottomAttachRow, work together to control the position of a drawing object in the exported Excel file. TopAttachRow specifies the Y coordinate of the upper left corner of the drawing object in the exported Excel file, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Float
Round box	Background	Specifies the background color of the object. Data type: String
	BorderColor	Specifies the color of the object's border. Data type: String
	BorderStyle	Specifies the line style of the object's border. Data type: Enumeration
	BorderWidth	Specifies the width of the object's border, in inches or centimeters. Data type: Float
	BottomAttachCol	The four Excel properties, TopAttachCol, TopAttachRow, BottomAttachCol and BottomAttachRow, work together to control the position of a drawing object in the exported Excel file. BottomAttachCol specifies the X coordinate of the lower right corner of the drawing object in the exported Excel file, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Float
	BottomAttachPosX	Specifies the horizontal coordinate of the bottom right point of the focus box in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	BottomAttachPosY	Specifies the vertical coordinate of the bottom right point of the focus box in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	BottomAttachRow	The four Excel properties, TopAttachCol, TopAttachRow, BottomAttachCol and BottomAttachRow, work together to control the position of a drawing object in the exported Excel file. BottomAttachRow specifies the Y coordinate of the lower right corner of the drawing object in the exported Excel file, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Float
	class	Specifies a CSS class to be applied to the object which is a valid class in the CSS file. Data type: String
	CornerFactor	Specifies the relative radius of the object's corners. Data type: Float
	ExportToDHTML	Specifies whether to include the object when exporting the report to Page Report Result. Data type: Boolean
	ExportToXLS	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
	ExternalCSSClassValue	Specifies a class selector to be applied to the object when exported as HTML. Data type: String
	ExternalID	It is mapped to the HTML attribute *id*. This attribute specifies a name to the object, which must be unique in the report. Data type: String
	ExternalStyle	It is mapped to the HTML attribute *style*. This attribute specifies style information for the object. Data type: String
	ExternalTitle	It is mapped to the HTML attribute *title*. This attribute offers advisory information about the object. Data type: String
	Invisible	Specifies whether to show the object in the design area and in the report result. All formulas and calculations will still be performed. Data type: Boolean
	Lang	It is mapped to the HTML attribute *lang*. This attribute specifies the base language of the object's attribute values and text content. Data type: String
	RecordLocation	Specifies the calculation point for the properties which use formulas. • **Default** - The values of the properties are calculated in the default location where they are located. • **Page Header** - The values of the properties are calculated in the page header. • **Page Footer** - The values of the properties are calculated in the page footer. Data type: String
	Style	The property can be used in two ways. • Specifies a style to be applied to the object. This works when a StyleGroup has been specified at the report sheet level and when there are styles in the StyleGroup that can be applied to the object. • Specifies a CSS selector to be applied to the object. Data type: String
	SuppressedIfNoRecords	Specifies whether to display the object in the report result when no record is returned to its parent data container. Data type: Boolean
	TopAttachCol	The four Excel properties, TopAttachCol, TopAttachRow, BottomAttachCol and BottomAttachRow, work together to control the position of a drawing object in the exported Excel file. TopAttachCol specifies the X coordinate of the upper left corner of the drawing object in the exported Excel file, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Float
	TopAttachPosX	Specifies the horizontal coordinate of the top left point of the focus box in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	TopAttachPosY	Specifies the vertical coordinate of the top left point of the focus box in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	TopAttachRow	The four Excel properties, TopAttachCol, TopAttachRow, BottomAttachCol and BottomAttachRow, work together to control the position of a drawing object in the exported Excel file. TopAttachRow specifies the Y coordinate of the upper left corner of the drawing object in the exported Excel file, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Float
Oval	Background	Specifies the background color of the object. Data type: String
	BorderColor	Specifies the color of the object's border. Data type: String
	BorderStyle	Specifies the line style of the object's border. Data type: Enumeration
	BorderWidth	Specifies the width of the object's border, in inches or centimeters. Data type: Float
	BottomAttachCol	The four Excel properties, TopAttachCol, TopAttachRow, BottomAttachCol and BottomAttachRow, work together to control the position of a drawing object in the exported Excel file. BottomAttachCol specifies the X coordinate of the lower right corner of the drawing object in the exported Excel file, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Float
	BottomAttachPosX	Specifies the horizontal coordinate of the bottom right point of the focus box in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	BottomAttachPosY	Specifies the vertical coordinate of the bottom right point of the focus box in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	BottomAttachRow	The four Excel properties, TopAttachCol, TopAttachRow, BottomAttachCol and BottomAttachRow, work together to control the position of a drawing object in the exported Excel file. BottomAttachRow specifies the Y coordinate of the lower right corner of the drawing object in the exported Excel file, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Float
	class	Specifies a CSS class to be applied to the object which is a valid class in the CSS file. Data type: String
	ExportToDHTML	Specifies whether to include the object when exporting the report to Page Report Result. Data type: Boolean
	ExportToXLS	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
	ExternalCSSClassValue	Specifies a class selector to be applied to the object when exported as HTML. Data type: String
	ExternalID	It is mapped to the HTML attribute *id*. This attribute specifies a name to the object, which must be unique in the report. Data type: String
	ExternalStyle	It is mapped to the HTML attribute *style*. This attribute specifies style information for the object. Data type: String
	ExternalTitle	It is mapped to the HTML attribute *title*. This attribute offers advisory information about the object. Data type: String
	Lang	It is mapped to the HTML attribute *lang*. This attribute specifies the base language of the object's attribute values and text content. Data type: String
	Invisible	Specifies whether to show the object in the design area and in the report result. All formulas and calculations will still be performed. Data type: Boolean
	RecordLocation	Specifies the calculation point for the properties which use formulas. • **Default** - The values of the properties are calculated in the default location where they are located. • **Page Header** - The values of the properties are calculated in the page header. • **Page Footer** - The values of the properties are calculated in the page footer. Data type: String
	Style	The property can be used in two ways. • Specifies a style to be applied to the object. This works when a StyleGroup has been specified at the report sheet level and when there are styles in the StyleGroup that can be applied to the object. • Specifies a CSS selector to be applied to the object. Data type: String
	SuppressedIfNoRecords	Specifies whether to display the object in the report result when no record is returned to its parent data container. Data type: Boolean
	TopAttachCol	The four Excel properties, TopAttachCol, TopAttachRow, BottomAttachCol and BottomAttachRow, work together to control the position of a drawing object in the exported Excel file. TopAttachCol specifies the X coordinate of the upper left corner of the drawing object in the exported Excel file, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Float
	TopAttachPosX	Specifies the horizontal coordinate of the top left point of the focus box in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	TopAttachPosY	Specifies the vertical coordinate of the top left point of the focus box in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	TopAttachRow	The four Excel properties, TopAttachCol, TopAttachRow, BottomAttachCol and BottomAttachRow, work together to control the position of a drawing object in the exported Excel file. TopAttachRow specifies the Y coordinate of the upper left corner of the drawing object in the exported Excel file, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Float
Arc	ArcAngle	Specifies the angle of the arc, in degrees. The value ranges from 0 to 360. As it gets closer to 360, the arc begins to take on an elliptical shape. Data type: Float
	Background	Specifies the background color of the object. Data type: String
	BorderColor	Specifies the color of the object's border. Data type: String
	BorderStyle	Specifies the line style of the object's border. Data type: Enumeration
	BorderWidth	Specifies the width of the object's border, in inches or centimeters. Data type: Float
	BottomAttachPosX	Specifies the horizontal coordinate of the bottom right point of the focus box in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	BottomAttachPosY	Specifies the vertical coordinate of the bottom right point of the focus box in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	class	Specifies a CSS class to be applied to the object which is a valid class in the CSS file. Data type: String
	ExportToDHTML	Specifies whether to include the object when exporting the report to Page Report Result. Data type: Boolean
	ExternalCSSClassValue	Specifies a class selector to be applied to the object when exported as HTML. Data type: String
	ExternalID	It is mapped to the HTML attribute *id*. This attribute specifies a name to the object, which must be unique in the report. Data type: String
	ExternalStyle	It is mapped to the HTML attribute *style*. This attribute specifies style information for the object. Data type: String
	ExternalTitle	It is mapped to the HTML attribute *title*. This attribute offers advisory information about the object. Data type: String
	Invisible	Specifies whether to show the object in the design area and in the report result. All formulas and calculations will still be performed. Data type: Boolean
	Lang	It is mapped to the HTML attribute *lang*. This attribute specifies the base language of the object's attribute values and text content. Data type: String
	RecordLocation	Specifies the calculation point for the properties which use formulas. • **Default** - The values of the properties are calculated in the default location where they are located. • **Page Header** - The values of the properties are calculated in the page header. • **Page Footer** - The values of the properties are calculated in the page footer. Data type: String
	StartAngle	Specifies the start angle of the arc, in degrees. The value ranges from 0 to 360. At 0, the vertex of the arc starts at 12:15 clock position. As it gets closer to 360, the arc travels in a counterclockwise motion. Data type: Float
	Style	The property can be used in two ways. • Specifies a style to be applied to the object. This works when a StyleGroup has been specified at the report sheet level and when there are styles in the StyleGroup that can be applied to the object. • Specifies a CSS selector to be applied to the object. Data type: String
	SuppressedIfNoRecords	Specifies whether to display the object in the report result when no record is returned to its parent data container. Data type: Boolean
	TopAttachPosX	Specifies the horizontal coordinate of the top left point of the focus box in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
	TopAttachPosY	Specifies the vertical coordinate of the top left point of the focus box in a banded panel. A drawing object can cross panels and the coordinate indicates the relative position in the involved panel. Data type: Float
Table	AnchorDispValue	Specifies a string or formula to display for the TOC entry for this object in the TOC Browser. The TOCAnchor property must be set to true for this property to take effect. Data type: String
	AutoExpands	Specifies whether the objects in cells of the table will expand to their empty neighbouring cells horizontally when the actual size of the objects is bigger than the cells due to no enough column or page width for auto fit. This property requires that the property AutoFit for the objects in the table cells is set to true. Data type: Boolean **Note:** The property does not apply to horizontal tables.
	Background	Specifies the background color of the object. Data type: String
	BorderColor	Specifies the color of the border of the object. Data type: String
	BottomLine	Specifies the line style of the bottom border of the object. Data type: Enumeration
	BorderWidth	Specifies the width of the border in inches or centimeters. Data type: Float
	Cache	Specifies whether to cache the dataset for this object in the data buffer so other objects which use the same dataset can share the data rather than perform their own SQL query. Data type: Boolean
	class	Specifies a CSS class to be applied to the object which is a valid class in the CSS file. Data type: String
	ExportToCSV	Specifies whether to include the object when exporting the report to CSV. Data type: Boolean
	ExportToDHTML	Specifies whether to include the object when exporting the report to Page Report Result. Data type: Boolean
	ExportToXLS	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
	ExternalCSSClassValue	Specifies a class selector to be applied to the object when exported as HTML. Data type: String
	ExternalID	It is mapped to the HTML attribute *id*. This attribute specifies a name to the object, which must be unique in the report. Data type: String
	ExternalStyle	It is mapped to the HTML attribute *style*. This attribute specifies style information for the object. Data type: String
	Height	Specifies the height of the object, in inches or centimeters. Data type: Float
	Horizontal	Specifies whether to show the table in horizontal style. Data type: Boolean
	Invisible	Specifies whether to show the object in the design area and in the report result. All formulas and calculations will still be performed if the property is set to true. Data type: Boolean
	LeftLine	Specifies the line style of the left border of the object. Data type: Enumeration
	position	Specifies the position of the object. • **static** - The object is positioned at the location in which it is inserted. The X, Y and other position-related properties will be hidden or disabled. The position of the object is affected by preceding objects in the same container. • **absolute** - The object will be located at the position specified by dragging and dropping or by setting its X and Y property values. The position of the object is not affected by preceding objects in the same container. Data type: String **Note:** This property only affects objects with a parent container that is the report body, a tabular cell, or a text box.
	RecordLocation	Specifies the calculation point for the properties which use formulas. • **Default** - The values of the properties are calculated in the default location where they are located. • **Page Header** - The values of the properties are calculated in the page header. • **Page Footer** - The values of the properties are calculated in the page footer. Data type: String
	RightLine	Specifies the line style of the right border of the object. Data type: Enumeration
	Style	The property can be used in two ways. • Specifies a style to be applied to the object. This works when a StyleGroup has been specified at the report sheet level and when there are styles in the StyleGroup that can be applied to the object. • Specifies a CSS selector to be applied to the object. Data type: String
	Summary	It is mapped to the HTML attribute *summary*. This attribute provides a summary of the object's purpose and structure. Data type: String
	Suppressed	Specifies whether to show the object in the design area and in the report result. All formulas and calculations will be skipped if the property is set to true. Data type: Boolean **Note:** When both the Invisible and Suppressed properties of an object are set to true, Suppressed has the higher priority.
	SuppressedIfNoRecords	Specifies whether to display the object in the report result when no record is returned to its parent data container. Data type: Boolean
	SuppressEmpty	Specifies whether to display the object in the report result when no record is returned to it. Data type: Boolean
	TOCAnchor	Specifies whether to include the object in the TOC Browser for the report. Data type: Boolean
	TopLine	Specifies the line style of the top border of the object. Data type: Enumeration
	Width	Specifies the width of the object, in inches or centimeters. Data type: Float
	X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property is ignored if the position property is set to static. Data type: Float
	Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property is ignored if the position property is set to static. Data type: Float
Table Column	AutoFit	Specifies whether to adjust the width of the object according to the contents. If the property is set to true, the width of the object will be adjusted automatically only when: • The position property for the field/label in this object is static or relative. • If the position property for the field/label in this object is absolute, then the AutoFit property for the field/label should be set to true. Data type: Boolean
	Invisible	Specifies whether to show the object in the design area and in the report result. All formulas and calculations will still be performed if the property is set to true. Data type: Boolean
Table Cell	Abbr	It is mapped to the HTML attribute *abbr*. This attribute specifies an abbreviated form of the content of the object. The abbreviated text may be rendered by user agents when appropriate in place of the object's content. Data type: String
	Axis	It is mapped to the HTML attribute *axis*. This attribute is used to place a cell into conceptual categories that can be considered to form axes in an n-dimensional space. The value of this attribute is a comma-separated list of category names. Data type: String
	Background	Specifies the background color of the object. Data type: String
	BorderColor	Specifies the color of the border of the object. Data type: String
	BorderWidth	Specifies the width of the border in inches or centimeters. Data type: Float
	BottomLine	Specifies the line style of the bottom border of the object. Data type: Enumeration
	class	Specifies a CSS class to be applied to the object which is a valid class in the CSS file. Data type: String
	ExportToCSV	Specifies whether to include the object when exporting the report to CSV. Data type: Boolean
	ExportToDHTML	Specifies whether to include the object when exporting the report to Page Report Result. Data type: Boolean
	ExportToXLS	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
	ExternalCSSClassValue	Specifies a class selector to be applied to the object when exported as HTML. Data type: String
	ExternalID	It is mapped to the HTML attribute *id*. This attribute specifies a name to the object, which must be unique in the report. Data type: String
	Headers	It is mapped to the HTML attribute *headers*. This attribute specifies the list of header cells that provide header information for the current data cell. The value of this attribute is a space-separated list of cell names; those cells must be named by setting their *id* attribute. Data type: String
	Height	Specifies the height of the object, in inches or centimeters. Data type: Float
	HorizontalAlignment	Specifies the horizontal justification of the content in the table cell. Data type: Enumeration **Note:** When the position property of the object in the table cell is set to absolute, this property does not take effect.
	JoiningMerge	Specifies whether to make all the sequential rows included in the cell vertically merged in the report result. It is meaningful to set the property only when the cell is merged with other cells in the same column. Data type: Boolean
	LeftLine	Specifies the line style of the left border of the object. Data type: Enumeration
	RecordLocation	Specifies the calculation point for the properties which use formulas. • **Default** - The values of the properties are calculated in the default location where they are located. • **Page Header** - The values of the properties are calculated in the page header. • **Page Footer** - The values of the properties are calculated in the page footer. Data type: String
	Repeat	Specifies whether to repeat the content of the cell in every page of the report result when the cell is split into pages. Available only for vertically merged cell. Data type: Boolean

	Bigfoot Gas	Gas type: the first number of the name of the Gas. Bigfoot
	Bigfoot Gas Mixture	Gas type: Mixture. Gas type: the first number of the name of the Gas. Bigfoot
	Bigfoot	Gas type: the first number of the name of the Gas. Bigfoot **Gas Mixture** - The name of the gas mixture. Bigfoot **Gas Mixture** - The name of the gas mixture. Bigfoot **Gas Mixture** - The name of the gas mixture

	Property	Description
	ColumnNumber	Specifies the number of columns which will be the object's width in the exported Excel or CSV file. The Columned property at the report tab level must be set to true for this property to take effect. Data type: String
	ExportToCSV	Specifies whether to include the object when exporting the report to CSV. Data type: Boolean
	ExportToDHTML	Specifies whether to include the object when exporting the report to Page Report Result. Data type: Boolean
	ExportToXLS	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
	ExternalCSSClassValue	Specifies a class selector to be applied to the object when exported as HTML. Data type: String
	ExternalID	A representation of the standard HTML attribute *id*. This attribute specifies a name to the object, which must be unique in the report. Data type: String
	ExternalTitle	A representation of the standard HTML attribute *title*. This attribute offers advisory information about the object. Data type: String
	GotoDetail	Specifies whether to show the detailed information when you click the object in Page Report Studio. This property is available only when the object is in the group header/footer panel of a banded object. Data type: Boolean
	GroupValueFormat	Specifies the encoding format for values on the category axis. Data type: String
	Height	Specifies the height of the object, in inches or centimeters. Data type: Float
	HintFilter	Specifies the number format for the hint message. Data type: String
	HintPercentFilter	Specifies the format for each hint percentage to the total. Data type: String
	HyperLink	Specifies whether to add a hyperlink that refers to another report or a website to the chart. Data type: String
	HyperLinkTarget	Specifies the target window or frame in which to display the content the Hyperlink property specifies. • **<Server Setting>** Loads the linked file according to setting of the Pop Up New Window for Links option in the Customize Profile panel of the Profile dialog in JReport Server. • **New Window** Loads the linked file into a new window. The window is not named. • **Whole Window** Loads the linked file into the full browser window. • **Same Frame** Loads the linked file into the same frame as the link. • **Parent Frame** Loads the linked file into the parent frame of the frame that contains the link. • **Other Frame** Loads the linked file into some other specified frame. If the frame name does not exist, the linked file will be loaded into a new window. Data type: String
	Invisible	Specifies whether to show the object in the design area and in the report result. All formulas and calculations will still be performed if the property is set to true. Data type: Boolean
	LogicColumn	Specifies whether to show the object in the next visible table cell in the same row when the column which holds the object is hidden. Data type: Enumeration **Notes:** • The next visible table cell should be completely empty, that is, the cell should not hold any content including blank space. • When the property for several objects in the same row is set to next visible column, and the columns holding these objects are all hidden, then only the object in the rightmost column will be shown in the next visible cell.
	LongDesc	A representation of the standard HTML attribute *longdesc*. This attribute specifies a link to a long description of the object. Data type: String
	MotionFilter	Specifies the data format for the motion field values displayed on the motion bar. Data type: String
	PatternList	Specifies patterns of the data markers. The patterns can be one or more of the following: Automatic, Color, Texture, and Gradient. Data type: String
	PlatformBackgroundBorderPenColor	Specifies the color of the platform border. Data type: String
	PlatformBackgroundBorderPenEndCaps	Specifies the ending style of the platform border line. Data type: Enumeration
	PlatformBackgroundBorderPenLineJoint	Specifies the joint style of the platform border line. Data type: Enumeration
	PlatformBackgroundBorderPenLineJointRadius	Specifies the radius for the joint of the platform border line. The property takes effect only when PlatformBackgroundBorderPenLineJoint is set to joint round. Data type: Float
	PlatformBackgroundBorderPenOutlined	Specifies whether to show the platform border line in Outline Path. If the property is set to true, the platform border line will be shown in Outline Path; otherwise, in Fill Path. Data type: Boolean
	PlatformBackgroundBorderPenStyle	Specifies the line style of the platform border. Data type: Enumeration
	PlatformBackgroundBorderPenThickness	Specifies the width of the platform border, in inches or centimeters. Data type: Float
	PlatformBackgroundBorderPenTransparency	Specifies the transparency of the platform border, in percent. Data type: Integer
	PlatformBackgroundBorderPenVariabledDash	Specifies whether to resize the dash automatically. • **true** - If selected, the dash size will be adjusted automatically, and the option Auto Adjust Dash in the Format Platform dialog will take effect. • **false** - If selected, the dash size will be of fixed size, and the option Fixed Dash Size in the Format Platform dialog will take effect. Data type: Boolean
	PlatformBackgroundBorderStyle	Specifies the border type of the platform. Data type: Enumeration
	PlatformBackgroundBottom	Specifies the left position of the background area, measured in a percentage of the chart width, from the left edge of the chart. Data type: Float
	PlatformBackgroundFillPatternColor	Specifies the color of the platform background, applied when the property PlatformBackgroundFillPatternType is set to color. Data type: String
	PlatformBackgroundFillPatternGradientEndColor	Specifies the color of the point where the gradient ends, applied when the property PlatformBackgroundFillPatternType is set to gradient. Data type: String
	PlatformBackgroundFillPatternGradientEndX	Specifies the horizontal position where the gradient ends, measured in a percentage of the platform width. Applied when the property PlatformBackgroundFillPatternType is set to gradient. Data type: Integer
	PlatformBackgroundFillPatternGradientEndY	Specifies the vertical position where the gradient ends, measured in a percentage of the platform height. Applied when the property PlatformBackgroundFillPatternType is set to gradient. Data type: Integer
	PlatformBackgroundFillPatternGradientStartColor	Specifies the color of the point where the gradient begins, applied when the property PlatformBackgroundFillPatternType is set to gradient. Data type: String
	PlatformBackgroundFillPatternGradientStartX	Specifies the horizontal position where the gradient begins, measured in a percentage of the platform width. Applied when the property PlatformBackgroundFillPatternType is set to gradient. Data type: Integer
	PlatformBackgroundFillPatternGradientStartY	Specifies the vertical position where the gradient begins, measured in a percentage of the platform height. Applied when the property PlatformBackgroundFillPatternType is set to gradient. Data type: Integer
	PlatformBackgroundFillPatternGradientStyle	Specifies the gradient style for the platform background, applied when the property PlatformBackgroundFillPatternType is set to gradient. Data type: Enumeration
	PlatformBackgroundFillPatternImageFile	Specifies the file name of the image, a portion of which will be displayed as the platform background. Applied when PlatformBackgroundFillPatternType is set to image. Enter the file name with suffix in the value cell. If the image is outside of the current catalog, you should include the full path of the image correctly. Data type: String
	PlatformBackgroundFillPatternImageHeight	The four properties, PlatformBackgroundFillPatternImageX, PlatformBackgroundFillPatternImageY, PlatformBackgroundFillPatternImageWidth, and PlatformBackgroundFillPatternImageHeight, together are used to define a portion of the image (the portion is a rectangle), which will be displayed as the platform background. They take effect when the property PlatformBackgroundFillPatternType is set to image. Image Height specifies the height of the image portion, measured in a percentage of the image height. Data type: Integer
	PlatformBackgroundFillPatternImageLayout	Specifies the layout style for the image that will be displayed as the platform background, applied when the property PlatformBackgroundFillPatternType is set to image. Data type: Enumeration
	PlatformBackgroundFillPatternImageWidth	The four properties, PlatformBackgroundFillPatternImageX, PlatformBackgroundFillPatternImageY, PlatformBackgroundFillPatternImageWidth, and PlatformBackgroundFillPatternImageHeight, together are used to define a portion of the image (the portion is a rectangle), which will be displayed as the platform background. They take effect when the property PlatformBackgroundFillPatternType is set to image. Image Width specifies the width of the image portion, measured in a percentage of the image width. Data type: Integer
	PlatformBackgroundFillPatternImageX	The four properties, PlatformBackgroundFillPatternImageX, PlatformBackgroundFillPatternImageY, PlatformBackgroundFillPatternImageWidth, and PlatformBackgroundFillPatternImageHeight, together are used to define a portion of the image (the portion is a rectangle), which will be displayed as the platform background. They take effect when the property PlatformBackgroundFillPatternType is set to image. Image X specifies the distance between the left border of image and the portion that will contain the pattern, measured in a percentage of the image width. Data type: Integer
	PlatformBackgroundFillPatternImageY	The four properties, PlatformBackgroundFillPatternImageX, PlatformBackgroundFillPatternImageY, PlatformBackgroundFillPatternImageWidth, and PlatformBackgroundFillPatternImageHeight, together are used to define a portion of the image (the portion is a rectangle), which will be displayed as the platform background. They take effect when the property PlatformBackgroundFillPatternType is set to image. Image Y specifies the distance between the top border of image and the portion that will contain the pattern, measured in a percentage of the image height. Data type: Integer
	PlatformBackgroundFillPatternTextureBackColor	Specifies the background color of the texture, applied when the property PlatformBackgroundFillPatternType is set to texture. Data type: String
	PlatformBackgroundFillPatternTextureForeColor	Specifies the foreground color of the texture, applied when the property PlatformBackgroundFillPatternType is set to texture. Data type: String
	PlatformBackgroundFillPatternTextureStyle	Specifies the texture style of the platform background, applied when the property PlatformBackgroundFillPatternType is set to texture. Data type: Enumeration
	PlatformBackgroundFillPatternTransparency	Specifies the transparency of the platform background, in percent. Data type: Integer
	PlatformBackgroundFillPatternType	Specifies the fill pattern for the platform background. Data type: Enumeration
	PlatformBackgroundLeft	Specifies the left position of the background area, measured in a percentage of the chart width, from the left edge of the chart. Data type: Float
	PlatformBackgroundRight	Specifies the right position of the background area, measured in a percentage of the chart width, from the right edge of the chart. Data type: Float
	PlatformBackgroundTop	Specifies the top position of the background area, measured in a percentage of the chart height, from the top edge of the chart. Data type: Float
	position	Specifies the position of the object. • **static** - The object is positioned at the location in which it is inserted. If selected, the X, Y and other position-related properties will be hidden or disabled. The position of the object is affected by preceding objects in the same container. • **absolute** - The object will be located at the position specified by dragging and dropping or by setting its X and Y property values. The position of the object is not affected by preceding objects in the same container. Data type: Enumeration **Note:** This property only affects objects with a parent container that is the report body, a table cell, a tabular cell, or a text box.
	PrimaryDataFilter	Specifies the data format for the primary value axis (the Y1 axis) to display the tick mark labels in the way you choose. Data type: String
	ReverseCategory	Specifies whether to reverse the sequence of the category field value. Data type: Boolean
	ReverseSeries	Specifies whether to reverse the sequence of the series field value. Data type: Boolean
	RowIndex	Specifies the Y coordinate of the object relative to its parent container when exported to Excel or CSV, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Integer
	RowNumber	Specifies the number of rows which will be the object's height in the exported Excel or CSV file. The Columned property at the report tab level must be set to true for this property to take effect. Data type: String
	SecondaryDataFilter	Specifies the data format for the secondary value axis (the Y2 axis) to display the tick mark labels in the way you choose. Data type: String
	SeriesEndOffset	The four properties work together to control the range of the data being displayed on a chart: CategoryStartOffset, CategoryEndOffset, SeriesStartOffset, and SeriesEndOffset. SeriesEndOffset specifies the ending offset of the series. Data type: Integer
	SeriesFilter	Specifies the data format for the series axis (the Z axis) to display the tick mark labels in the way you choose. Data type: String
	SeriesStartOffset	The four properties work together to control the range of the data being displayed on a chart: CategoryStartOffset, CategoryEndOffset, SeriesStartOffset, and SeriesEndOffset. SeriesStartOffset specifies the starting offset of the series. Data type: Integer
	SeriesValueEncoding	Specifies the encoding format for values on the series axis. Data type: String
	ShowLegend	Specifies whether to make the legend visible in a chart. Data type: Boolean
	SortCategory	Specifies the sorting order for the category field values. Data type: Enumeration
	SortSeries	Specifies the sorting order for the series field values. Data type: Enumeration
	Style	The property can be used in two ways. • Specifies a style to be applied to the object. This works when a StyleGroup has been specified at the report sheet level and when there are styles in the StyleGroup that can be applied to the object. • Specifies a CSS selector to be applied to the object. Data type: String
	Suppressed	Specifies whether to show the object in the design area and in the report result. All formulas and calculations will be skipped if the property is set to true. Data type: Boolean **Note:** When both the Invisible and Suppressed properties of an object are set to true, Suppressed has the higher priority.
	SuppressedIfNoRecords	Specifies whether to display the object in the report result when no record is returned to its parent data container. Data type: Boolean
	SwapGroup	Specifies to display values from different data fields by switching data between the category and series axes, the category and value axes of a chart. By default, this property is set to false, which means no switch will take place in the chart. • **Switching between the category and series axes** If you create a chart by adding fields to all of its axes, and set SwapGroup to true, data on the category and series axes will be switched. • **Switching between the category and value axes** If you create a chart by adding one field to the category axis and several to the value axis, and set SwapGroup to true, data on the category and value axes will be switched. Data type: Boolean **Note:** If the chart has only one field on the category axis and one on the value axis, even though the property value is set to true, there will be no switch take place on the two axes.
	TOCAnchor	Specifies whether to include the object in the TOC Browser for the report. Data type: Boolean
	Width	Specifies the width of the object, in inches or centimeters. Data type: Float
	X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property is ignored if the position property is set to static. Data type: Float
	XHyperLink	Specifies whether to add a hyperlink that refers to another report or a website to the X axis labels. Data type: String
	XHyperLinkTarget	Specifies the target window or frame in which to display the content the XHyperlink property specifies. • **<Server Setting>** Loads the linked file according to setting of the Pop Up New Window for Links option in the Customize Profile panel of the Profile dialog in JReport Server. • **New Window** Loads the linked file into a new window. The window is not named. • **Whole Window** Loads the linked file into the full browser window. • **Same Frame** Loads the linked file into the same frame as the link. • **Parent Frame** Loads the linked file into the parent frame of the frame that contains the link. • **Other Frame** Loads the linked file into some other specified frame. If the frame name does not exist, the linked file will be loaded into a new window. Data type: String
	Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property is ignored if the position property is set to static. Data type: Float
	ZHyperLink	Specifies whether to add a hyperlink that refers to another report or a website to the Z axis labels. Data type: String
	ZHyperLinkTarget	Specifies the target window or frame in which to display the content the ZHyperlink property specifies. • **<Server Setting>** Loads the linked file according to setting of the Pop Up New Window for Links option in the Customize Profile panel of the Profile dialog in JReport Server. • **New Window** Loads the linked file into a new window. The window is not named. • **Whole Window** Loads the linked file into the full browser window. • **Same Frame** Loads the linked file into the same frame as the link. • **Parent Frame** Loads the linked file into the parent frame of the frame that contains the link. • **Other Frame** Loads the linked file into some other specified frame. If the frame name does not exist, the linked file will be loaded into a new window. Data type: String
Chart Paper	Background	Specifies the properties of the chart paper background.
	BorderColor	Specifies the color of the paper border. Data type: String
	BorderWidth	Specifies the width of the paper border, in inches or centimeters. Data type: Float
	BottomLine	Specifies the line style of the bottom border of the object. Data type: Enumeration
	class	Specifies a CSS class to be applied to the object which is a valid class in the CSS file. Data type: String
	ColumnIndex	Specifies the X coordinate of the object relative to its parent container when exported to Excel or CSV, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Integer
	ExportToCSV	Specifies whether to include the object when exporting the report to CSV. Data type: Boolean
	ExportToDHTML	Specifies whether to include the object when exporting the report to Page Report Result. Data type: Boolean
	ExportToXLS	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
	ExternalCSSClassValue	Specifies a class selector to be applied to the object when exported as HTML. Data type: String
	Height	Specifies the height of the object, in inches or centimeters. Data type: Float
	Invisible	Specifies whether to show the object in the design area and in the report result. All formulas and calculations will still be performed if the property is set to true. Data type: Boolean
	LeftLine	Specifies the line style of the left border of the object. Data type: Enumeration
	position	Specifies the position of the object. • **static** - The object is positioned at the location in which it is inserted. The X, Y and other position-related properties will be hidden or disabled. The position of the object is affected by preceding objects in the same container. • **absolute** - The object will be located at the position specified by dragging and dropping or by setting its X and Y property values. The position of the object is not affected by preceding objects in the same container. Data type: String **Note:** This property only affects objects with a parent container that is the report body, a tabular cell, or a text box.
	RecordLocation	Specifies the calculation point for the properties which use formulas. • **Default** - The values of the properties are calculated in the default location where they are located. • **Page Header** - The values of the properties are calculated in the page header. • **Page Footer** - The values of the properties are calculated in the page footer. Data type: String
	RightLine	Specifies the line style of the right border of the object. Data type: Enumeration
	RowIndex	Specifies the Y coordinate of the object relative to its parent container when exported to Excel or CSV, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Integer
	Style	The property can be used in two ways. • Specifies a style to be applied to the object. This works when a StyleGroup has been specified at the report sheet level and when there are styles in the StyleGroup that can be applied to the object, or type in the style name. • Specifies a CSS selector to be applied to the object. Data type: String
	Suppressed	Specifies whether to show the object in the design area and in the report result. All formulas and calculations will be skipped if the property is set to true. Data type: Boolean **Note:** When both the Invisible and Suppressed properties of an object are set to true, Suppressed has the higher priority.
	SuppressedIfNoRecords	Specifies whether to display the object in the report result when no record is returned to its parent data container. Data type: Boolean
	TopLine	Specifies the line style of the top border of the object. Data type: Enumeration
	Width	Specifies the width of the object, in inches or centimeters. Data type: Float
	X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property takes effect only when the Placement property for the chart legend is set to auto. Data type: Float
	Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property takes effect only when the Placement property for the chart legend is set to auto. Data type: Float
Chart legend	AutoSize	Specifies whether to resize the legend automatically. Data type: Boolean
	BottomMargin	Specifies the distance in inches or centimeters between the legend labels and the bottom border of the legend. Data type: Float
	class	Specifies a CSS class to be applied to the object which is a valid class in the CSS file. Data type: String
	ExternalCSSClassValue	Specifies a class selector to be applied to the object when exported as HTML. Data type: String
	Height	Specifies the height of the object, in inches or centimeters. Data type: Float
	IconAlignment	Specifies the relative position of the icon to the legend. Data type: Enumeration
	IconBorderPenColor	Specifies the color of the icon borders. Data type: String
	IconBorderPenEndCaps	Specifies the ending style of the icon border line. Data type: Enumeration
	IconBorderPenLineJoint	Specifies the joint style of the icon border line. Data type: Enumeration
	IconBorderPenOutlined	Specifies whether to show the icon border line in Outline Path. If the property is set to true, the icon border line will be shown in Outline Path; otherwise, in Fill Path. Data type: Boolean
	IconBorderPenStyle	Specifies the line style of the icon borders. Data type: Enumeration
	IconBorderPenThickness	Specifies the width of the icon borders in inches or centimeters. Data type: Float
	IconBorderPenTransparency	Specifies the transparency of the icon borders, in percent. Data type: Integer
	IconBorderPenVariabledDash	Specifies whether to resize the dash automatically. If the property is set to true, the dash size will be adjusted automatically; otherwise, the dash size will be of fixed size. Data type: Boolean
	IconHeight	Specifies the height of the icon, in inches or centimeters. Data type: Float
	IconTextGap	Specifies the distance between each legend label and icon. Data type: Float
	IconWidth	Specifies the width of the icon, in inches or centimeters. Data type: Float
	LabelFilterSource	Specifies the data format option for the chart legend labels. • **Axis Data Format** - The data format of the legend labels will follow that of the axis data: if the chart has series axis, the legend label format will be the same as the series axis data format; if not, the category axis data format will be used to the legend labels. • **Legend Data Format** - The data format of the legend labels will be controlled by the property LegendFilter. Data type: Enumeration
	LabelHSpacing	Specifies the horizontal distance between two adjacent legend labels. Data type: Float
	LabelTextFontName	Specifies the font face for the legend labels. Data type: Enumeration
	LabelTextFontSize	Specifies the font size for the legend labels. Data type: Integer
	LabelTextFontStyle	Specifies the font style for the legend labels. Data type: Enumeration
	LabelTextPenStyle	Specifies the style for the label text outline. Data type: Enumeration
	LabelTextPatternColor	Specifies the color of the label, applied when the property LabelTextPatternType is set to color. Data type: String
	LabelTextPatternGradientEndColor	Specifies the color of the point where the gradient ends, applied when the property LabelTextPatternType is set to gradient. Data type: String
	LabelTextPatternGradientEndX	Specifies the horizontal position where the gradient ends, measured in a percentage of the label text width. Applied when the property LabelTextPatternType is set to gradient. Data type: Integer
	LabelTextPatternGradientEndY	Specifies the vertical position where the gradient ends, measured in a percentage of the label text height. Applied when the property LabelTextPatternType is set to gradient. Data type: Integer
	LabelTextPatternGradientStartColor	Specifies the color of the point where the gradient begins, applied when the property LabelTextPatternType is set to gradient. Data type: String
	LabelTextPatternGradientStartX	Specifies the horizontal position where the gradient begins, measured in a percentage of the label text width. Applied when the property LabelTextPatternType is set to gradient. Data type: Integer
	LabelTextPatternGradientStartY	Specifies the vertical position where the gradient begins, measured in a percentage of the label text height. Applied when the property LabelTextPatternType is set to gradient. Data type: Integer
	LabelTextPatternGradientStyle	Specifies the gradient style for the label, applied when the property LabelTextPatternType is set to gradient. The image fill type does not apply to the legend label. Data type: Enumeration
	LabelTextPatternTextureBackColor	Specifies the background color of the texture, applied when the property LabelTextPatternType is set to texture. Data type: String
	LabelTextPatternTextureForeColor	Specifies the foreground color of the texture, applied when the property LabelTextPatternType is set to texture. Data type: String
	LabelTextPatternTextureStyle	Specifies the texture style for the label, applied when the property LabelTextPatternType is set to texture. Data type: Enumeration
	LabelTextPatternTransparency	Specifies the transparency of the background, in percent. Data type: Integer
	LabelTextPatternType	Specifies the fill pattern for the label. Data type: Enumeration
	LabelTextPenColor	Specifies the color for the label text outline. Data type: String
	LabelTextPenEndCaps	Specifies the ending style for the label text outline. Data type: Enumeration
	LabelTextPenLineJoint	Specifies the joint style for the label text outline. Data type: Enumeration
	LabelTextPenOutlined	Specifies whether to show the label text outline in outline form. If the property is set to true, the outline will be shown in outline form; otherwise, in whole form. Data type: Boolean
	LabelTextPenThickness	Specifies the thickness for the label text outline. Data type: Float
	LabelTextPenTransparency	Specifies the transparency for the label text outline, in percent. Data type: Integer
	LabelTextPenVariabledDash	Specifies whether to resize the dash automatically. If the property is set to true, the dash size will be adjusted automatically; otherwise, the dash size will be of fixed size. Data type: Boolean
	LabelTextRotation	Specifies the rotation angle of each legend label around its center, in degrees. The default value is 0. Data type: Float
	LabelTextShearing	Specifies the shearing transformation of each legend label around its center. The default value is 0. Data type: Float
	LabelTextSpecialEffect1	Specifies the special effect for the label text. Data type: Enumeration
	LabelTextSpecialEffect2	Specifies whether the label text will in superscript or subscript form, or neither of them. Data type: Enumeration
	LabelTextStrikethrough	Specifies the style of the line by which the label texts are struck through. Data type: Enumeration
	LabelTextUnderline	Specifies the underline style for the legend labels. Data type: Enumeration
	LabelTextWordWrap	Specifies whether to enable the word wrap function for the label text, not applying to radar charts, bubble charts, scatter charts and gauge charts. If the text contains special characters (such as ",", "." and space), it will be broken at the position of one of the special characters. Data type: Boolean
	LabelVSpacing	Specifies the vertical distance between two adjacent legend labels. Data type: Float
	LeftMargin	Specifies the distance in inches or centimeters between the legend labels and the left border of the legend. Data type: Float
	LegendBackgroundBorderPenColor	Specifies the color of the legend border. Data type: String
	LegendBackgroundBorderPenEndCaps	Specifies the ending style of the legend border line. Data type: Enumeration
	LegendBackgroundBorderPenLineJoint	Specifies the joint style of the legend border line. Data type: Enumeration
	LegendBackgroundBorderPenOutlined	Specifies whether to show the legend border line in Outline Path. If the property is set to true, the legend border line will be shown in Outline Path; otherwise, in Fill Path. Data type: Boolean
	LegendBackgroundBorderPenStyle	Specifies the line style of the legend border. Data type: Enumeration
	LegendBackgroundBorderPenThickness	Specifies the width of the legend border, in inches or centimeters. Data type: Float
	LegendBackgroundBorderPenTransparency	Specifies the transparency of the legend border, in percent. Data type: Integer
	LegendBackgroundBorderPenVariabledDash	Specifies whether to resize the dash automatically. • **true** - If selected, the dash size will be adjusted automatically, and the option Auto Adjust Dash in the Format Legend dialog will take effect. • **false** - If selected, the dash size will be of fixed size, and the option Fixed Dash Size in the Format Legend dialog will take effect. Data type: Boolean
	LegendBackgroundBorderStyle	Specifies the border type of the legend. Data type: Enumeration
	LegendBackgroundBottom	Specifies the bottom position of the background area, measured in a percentage of the legend height, from the bottom edge of the legend. Data type: Float
	LegendBackgroundFillPatternColor	Specifies the color of the legend background, applied when the property LegendBackgroundFillPatternType is set to color. Data type: String
	LegendBackgroundFillPatternGradientEndColor	Specifies the color of the point where the gradient ends, applied when the property LegendBackgroundFillPatternType is set to gradient. Data type: String
	LegendBackgroundFillPatternGradientEndX	Specifies the horizontal position where the gradient ends, measured in a percentage of the legend width. Applied when the property LegendBackgroundFillPatternTypee is set to gradient. Data type: Integer
	LegendBackgroundFillPatternGradientEndY	Specifies the vertical position where the gradient ends, measured in a percentage of the legend height. Applied when the property LegendBackgroundFillPatternType is set to gradient. Data type: Integer
	LegendBackgroundFillPatternGradientStartColor	Specifies the color of the point where the gradient begins, applied when the property LegendBackgroundFillPatternType is set to gradient. Data type: String
	LegendBackgroundFillPatternGradientStartX	Specifies the horizontal position where the gradient begins, measured in a percentage of the legend width. Applied when the property LegendBackgroundFillPatternType is set to gradient. Data type: Integer
	LegendBackgroundFillPatternGradientStartY	Specifies the vertical position where the gradient begins, measured in a percentage of the legend height. Applied when the property LegendBackgroundFillPatternType is set to gradient. Data type: Integer
	LegendBackgroundFillPatternGradientStyle	Specifies the gradient style for the legend background, applied when the property LegendBackgroundFillPatternType is set to gradient. Data type: Enumeration
	LegendBackgroundFillPatternImageFile	Specifies the file name of the image, a portion of which will be displayed as the legend background. Applied when the property LegendBackgroundFillPatternType is set to image. Enter the file name with suffix in the value cell. If the image is outside of the current catalog, you should include the full path of the image correctly. Data type: String
	LegendBackgroundFillPatternImageHeight	The four properties, LegendBackgroundFillPatternImageX, LegendBackgroundFillPatternImageY, LegendBackgroundFillPatternImageWidth, and LegendBackgroundFillPatternImageHeight, together are used to define a portion of the image (the portion is a rectangle), which will be displayed as the legend background. They take effect when the property LegendBackgroundFillPatternType is set to image. LegendBackgroundFillPatternImageHeight specifies the height of the image portion, measured in a percentage of the image height. Data type: Integer
	LegendBackgroundFillPatternImageLayout	Specifies the layout style for the image that will be displayed as the legend background, applied when the property LegendBackgroundFillPatternType is set to image. Data type: Enumeration
	LegendBackgroundFillPatternImageWidth	The four properties, LegendBackgroundFillPatternImageX, LegendBackgroundFillPatternImageY, LegendBackgroundFillPatternImageWidth, and LegendBackgroundFillPatternImageHeight, together are used to define a portion of the image (the portion is a rectangle), which will be displayed as the legend background. They take effect when the property LegendBackgroundFillPatternType is set to image. LegendBackgroundFillPatternImageWidth specifies the width of the image portion, measured in a percentage of the image width. Data type: Integer
	LegendBackgroundFillPatternImageX	The four properties, LegendBackgroundFillPatternImageX, LegendBackgroundFillPatternImageY, LegendBackgroundFillPatternImageWidth, and LegendBackgroundFillPatternImageHeight, together are used to define a portion of the image (the portion is a rectangle), which will be displayed as the legend background. They take effect when the property LegendBackgroundFillPatternType is set to image. LegendBackgroundFillPatternImageX specifies the distance between the left border of image and the portion that will contain the pattern, measured in a percentage of the image width. Data type: Integer
	LegendBackgroundFillPatternImageY	The four properties, LegendBackgroundFillPatternImageX, LegendBackgroundFillPatternImageY, LegendBackgroundFillPatternImageWidth, and LegendBackgroundFillPatternImageHeight, together are used to define a portion of the image (the portion is a rectangle), which will be displayed as the legend background. They take effect when the property Fill Type is set to image. LegendBackgroundFillPatternImageY specifies the distance between the top border of image and the portion that will contain the pattern, measured in a percentage of the image height. Data type: Integer
	LegendBackgroundFillPatternTextureBackColor	Specifies the background color of the texture, applied when the property LegendBackgroundFillPatternType is set to texture. Data type: String
	LegendBackgroundFillPatternTextureForeColor	Specifies the foreground color of the texture, applied when the property LegendBackgroundFillPatternType is set to texture. Data type: String
	LegendBackgroundFillPatternTextureStyle	Specifies the texture style of the legend background, applied when the property LegendBackgroundFillPatternType is set to texture. Data type: Enumeration
	LegendBackgroundFillPatternTransparency	Specifies the transparency of the legend background, in percent. Data type: Integer
	LegendBackgroundFillPatternType	Specifies the fill pattern for the legend background. Data type: Enumeration
	LegendBackgroundLeft	Specifies the left position of the background area, measured in a percentage of the legend width, from the left edge of the legend. Data type: Float
	LegendBackgroundRight	Specifies the right position of the background area, measured in a percentage of the legend width, from the right edge of the legend. Data type: Float
	LegendBackgroundTop	Specifies the top position of the background area, measured in a percentage of the legend height, from the top edge of the legend. Data type: Float
	LegendFilter	Specifies the data format of the legend labels, applied when LabelFilterSource is set to Legend Data Format. Data type: String
	PercentFormat	Specifies the format of the percentage values on the legend. Applied when ShowPercent is set to true. Data type: String
	Placement	Specifies the position of the legend in the platform. Data type: Enumeration
	ReverseLabels	Specifies whether to reverse the order of the labels in the legend object. Data type: Boolean
	RightMargin	Specifies the distance in inches or centimeters between the legend labels and the right border of the legend. Data type: Float
	ShowPercent	Specifies whether to show the percentage of each entry value to the total on the legend. Applies to pie, bar and bench charts that have only one value axis. Data type: Boolean
	ShowScrollbar	Specifies whether to show a scrollbar on the legend to fully view the legend content when the content does not fit into the legend. Data type: Boolean
	ShowTips	Specifies whether to show the corresponding data information when the mouse pointer points at a target in the chart legend in Designer view mode, in HTML result, in Page Report Studio, or in Web Report Studio or JDashboard. Data type: Boolean
	ShowValues	Specifies whether to show the values for the legend entries. Applies to pie, bar and bench charts that have only one value axis. Data type: Boolean
	Style	The property can be used in two ways. • Specifies a style to be applied to the object. This works when a StyleGroup has been specified at the report sheet level and when there are styles in the StyleGroup that can be applied to the object, or type in the style name. • Specifies a CSS selector to be applied to the object. Data type: String
	SubPlacement	Specifies the position of the legend further based on the setting of the property Placement. Data type: Enumeration
	TopMargin	Specifies the distance in inches or centimeters between the legend labels and the top border of the legend. Data type: Float
	TotalValueText	Specifies the text of the total value label on the legend. Applies to pie, bar and bench charts that have only one value axis. Data type: String
	Truncate	Specifies whether to truncate the legend entry label text when the text overflow the labels. Data type: Boolean
	ValueFormat	Specifies the number format of the legend entry values. Applied when ShowValues is set to true. Data type: String
	Width	Specifies the width of the object, in inches or centimeters. Data type: Float
	X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property takes effect only when the Placement property is set to auto. Data type: Float
	Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property takes effect only when the Placement property is set to auto. Data type: Float
Chart Label	Alt	Specifies the alternate text of the image, which will be shown when the image cannot be displayed. This property applies only when the chart is exported to HTML as an image chart. Data type: String
	AnchorDispValue	Specifies a string or formula to display for the TOC entry for this object in the TOC Browser. The TOCAnchor property must be set to true for this property to take effect. Data type: String
	AntiAliasing	Specifies whether to make the edges in a chart smooth. Data type: Boolean
	Cache	Specifies whether to cache the dataset for this object in the data buffer so other objects which use the same dataset can share the data rather than perform their own SQL query. Data type: Boolean
	CategoryEndOffset	The four properties work together to control the range of the data being displayed on a chart: CategoryStartOffset, CategoryEndOffset, SeriesStartOffset, and SeriesEndOffset. CategoryEndOffset specifies the ending offset of the categories. It does not apply to the charts which involve only one group. Data type: Integer
	CategoryFilter	Specifies the data format for the category axis (the X axis) to display the tick mark labels in the way you choose. Data type: String

CategoryStartOffset	The four properties work together to control the range of the data being displayed on a chart: CategoryStartOffset, CategoryEndOffset, SeriesStartOffset, and SeriesEndOffset. CategoryStartOffset specifies the starting offset of the categories. It does not apply to the charts which involve only one group. Data type: Integer
class	Specifies a CSS class to be applied to the object which is a valid class in the CSS file. Data type: String
ColumnIndex	Specifies the X coordinate of the object relative to its parent container when exported to Excel or CSV, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Integer
ColumnNumber	Specifies the number of columns which will be the object's width in the exported Excel or CSV file. The Columned property at the report tab level must be set to true for this property to take effect. Data type: String
ExportToCSV	Specifies whether to include the object when exporting the report to CSV. Data type: Boolean
ExportToDHTML	Specifies whether to include the object when exporting the report to Page Report Result. Data type: Boolean
ExportToXLS	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
ExternalCSSClassValue	Specifies a class selector to be applied to the object when exported as HTML. Data type: String
ExternalID	It is mapped to the HTML attribute *id*. This attribute specifies a name to the object, which must be unique in the report. Data type: String
ExternalTitle	It is mapped to the HTML attribute *title*. This attribute offers advisory information about the object. Data type: String
GotoDetail	Specifies whether to show the detailed information when you click the object in Page Report Studio. This property is available only when the object is in the group header/footer panel of a banded object. Data type: Boolean
GroupValueFormat	Specifies the encoding format for values on the category axis. Data type: String
Height	Specifies the height of the object, in inches or centimeters. Data type: Float
HintFilter	Specifies the number format for the hint message. Data type: String
HintPercentFilter	Specifies the format for each hint percentage to the total. Data type: String
HyperLink	Specifies whether to add a hyperlink that refers to another report or a website to the chart. Data type: String
HyperLinkTarget	Specifies the target window or frame in which to display the content the Hyperlink property specifies. • **<Server Setting>** Loads the linked file according to setting of the Pop Up New Window for Links option in the Customize Profile panel of the Profile dialog in JReport Server. • **New Window** Loads the linked file into a new window. The window is not named. • **Whole Window** Loads the linked file into the full browser window. • **Same Frame** Loads the linked file into the same frame as the link. • **Parent Frame** Loads the linked file into the parent frame of the frame that contains the link. • **Other Frame** Loads the linked file into some other specified frame. If the frame name does not exist, the linked file will be loaded into a new window. Data type: String
Invisible	Specifies whether to show the object in the design area and in the report result. All formulas and calculations will still be performed if the property is set to true. Data type: Boolean
LogicColumn	Specifies whether to show the object in the next visible table cell in the same row when the column which holds the object is hidden. Data type: Enumeration **Notes:** • The next visible table cell should be completely empty, that is, the cell should not hold any content including blank space. • When the property for several objects in the same row is set to next visible column, and the columns holding these objects are all hidden, then only the object in the rightmost column will be shown in the next visible cell.
LongDesc	A representation of the standard HTML attribute *longdesc*. This attribute specifies a link to a long description of the object. Data type: String
MotionFilter	Specifies the data format for the motion field values displayed on the motion bar. Data type: String
PatternList	Specifies patterns of the data markers. The patterns can be one or more of the following: Automatic, Color, Texture, and Gradient. Data type: String
PlatformBackgroundBorderPenColor	Specifies the color of the platform border. Data type: String
PlatformBackgroundBorderPenEndCaps	Specifies the ending style of the platform border line. Data type: Enumeration
PlatformBackgroundBorderPenLineJoint	Specifies the joint style of the platform border line. Data type: Enumeration
PlatformBackgroundBorderPenLineJointRadius	Specifies the radius for the joint of the platform border line. The property takes effect only when PlatformBackgroundBorderPenLineJoint is set to joint round. Data type: Float
PlatformBackgroundBorderPenOutlined	Specifies whether to show the platform border line in Outline Path. If the property is set to true, the platform border line will be shown in Outline Path; otherwise, in Fill Path. Data type: Boolean
PlatformBackgroundBorderPenStyle	Specifies the line style of the platform border. Data type: Enumeration
PlatformBackgroundBorderPenThickness	Specifies the width of the platform border, in inches or centimeters. Data type: Float
PlatformBackgroundBorderPenTransparency	Specifies the transparency of the platform border, in percent. Data type: Integer
PlatformBackgroundBorderPenVariabledDash	Specifies whether to resize the dash automatically. • **true** - If selected, the dash size will be adjusted automatically, and the option Auto Adjust Dash in the Format Platform dialog will take effect. • **false** - If selected, the dash size will be of fixed size, and the option Fixed Dash Size in the Format Platform dialog will take effect. Data type: Boolean
PlatformBackgroundBorderStyle	Specifies the border type of the platform. Data type: Enumeration
PlatformBackgroundBottom	Specifies the left position of the background area, measured in a percentage of the chart width, from the left edge of the chart. Data type: Float
PlatformBackgroundFillPatternColor	Specifies the color of the platform background, applied when the property PlatformBackgroundFillPatternType is set to color. Data type: String
PlatformBackgroundFillPatternGradientEndColor	Specifies the color of the point where the gradient ends, applied when the property PlatformBackgroundFillPatternType is set to gradient. Data type: String
PlatformBackgroundFillPatternGradientEndX	Specifies the horizontal position where the gradient ends, measured in a percentage of the platform width. Applied when the property PlatformBackgroundFillPatternType is set to gradient. Data type: Integer
PlatformBackgroundFillPatternGradientEndY	Specifies the vertical position where the gradient ends, measured in a percentage of the platform height. Applied when the property PlatformBackgroundFillPatternType is set to gradient. Data type: Integer
PlatformBackgroundFillPatternGradientStartColor	Specifies the color of the point where the gradient begins, applied when the property PlatformBackgroundFillPatternType is set to gradient. Data type: String
PlatformBackgroundFillPatternGradientStartX	Specifies the horizontal position where the gradient begins, measured in a percentage of the platform width. Applied when the property PlatformBackgroundFillPatternType is set to gradient. Data type: Integer
PlatformBackgroundFillPatternGradientStartY	Specifies the vertical position where the gradient begins, measured in a percentage of the platform height. Applied when the property PlatformBackgroundFillPatternType is set to gradient. Data type: Integer
PlatformBackgroundFillPatternGradientStyle	Specifies the gradient style for the platform background, applied when the property PlatformBackgroundFillPatternType is set to gradient. Data type: Enumeration
PlatformBackgroundFillPatternImageFile	Specifies the file name of the image, a portion of which will be displayed as the platform background. Applied when PlatformBackgroundFillPatternType is set to image. Enter the file name with suffix in the value cell. If the image is outside of the current catalog, you should include the full path of the image correctly. Data type: String
PlatformBackgroundFillPatternImageHeight	The four properties, PlatformBackgroundFillPatternImageX, PlatformBackgroundFillPatternImageY, PlatformBackgroundFillPatternImageWidth, and PlatformBackgroundFillPatternImageHeight, together are used to define a portion of the image (the portion is a rectangle), which will be displayed as the platform background. They take effect when the property PlatformBackgroundFillPatternType is set to image. Image Height specifies the height of the image portion, measured in a percentage of the image height. Data type: Integer
PlatformBackgroundFillPatternImageLayout	Specifies the layout style for the image that will be displayed as the platform background, applied when the property PlatformBackgroundFillPatternType is set to image. Data type: Enumeration
PlatformBackgroundFillPatternImageWidth	The four properties, PlatformBackgroundFillPatternImageX, PlatformBackgroundFillPatternImageY, PlatformBackgroundFillPatternImageWidth, and PlatformBackgroundFillPatternImageHeight, together are used to define a portion of the image (the portion is a rectangle), which will be displayed as the platform background. They take effect when the property PlatformBackgroundFillPatternType is set to image. Image Width specifies the width of the image portion, measured in a percentage of the image width. Data type: Integer
PlatformBackgroundFillPatternImageX	The four properties, PlatformBackgroundFillPatternImageX, PlatformBackgroundFillPatternImageY, PlatformBackgroundFillPatternImageWidth, and PlatformBackgroundFillPatternImageHeight, together are used to define a portion of the image (the portion is a rectangle), which will be displayed as the platform background. They take effect when the property PlatformBackgroundFillPatternType is set to image. Image X specifies the distance between the left border of image and the portion that will contain the pattern, measured in a percentage of the image width. Data type: Integer
PlatformBackgroundFillPatternImageY	The four properties, PlatformBackgroundFillPatternImageX, PlatformBackgroundFillPatternImageY, PlatformBackgroundFillPatternImageWidth, and PlatformBackgroundFillPatternImageHeight, together are used to define a portion of the image (the portion is a rectangle), which will be displayed as the platform background. They take effect when the property PlatformBackgroundFillPatternType is set to image. Image Y specifies the distance between the top border of image and the portion that will contain the pattern, measured in a percentage of the image height. Data type: Integer
PlatformBackgroundFillPatternTextureBackColor	Specifies the background color of the texture, applied when the property PlatformBackgroundFillPatternType is set to texture. Data type: String
PlatformBackgroundFillPatternTextureForeColor	Specifies the foreground color of the texture, applied when the property PlatformBackgroundFillPatternType is set to texture. Data type: String
PlatformBackgroundFillPatternTextureStyle	Specifies the texture style of the platform background, applied when the property PlatformBackgroundFillPatternType is set to texture. Data type: Enumeration
PlatformBackgroundFillPatternTransparency	Specifies the transparency of the platform background, in percent. Data type: Integer
PlatformBackgroundFillPatternType	Specifies the fill pattern for the platform background. Data type: Enumeration
PlatformBackgroundLeft	Specifies the left position of the background area, measured in a percentage of the chart width, from the left edge of the chart. Data type: Float
PlatformBackgroundRight	Specifies the right position of the background area, measured in a percentage of the chart width, from the right edge of the chart. Data type: Float
PlatformBackgroundTop	Specifies the top position of the background area, measured in a percentage of the chart height, from the top edge of the chart. Data type: Float
position	Specifies the position of the object. • **static** - The object is positioned at the location in which it is inserted. If selected, the X, Y and other position-related properties will be hidden or disabled. The position of the object is affected by preceding objects in the same container. • **absolute** - The object will be located at the position specified by dragging and dropping or by setting its X and Y property values. The position of the object is not affected by preceding objects in the same container. Data type: Enumeration **Note:** This property only affects objects with a parent container that is the report body, a table cell, a tabular cell, or a text box.
PrimaryDataFilter	Specifies the data format for the primary value axis (the Y1 axis) to display the tick mark labels in the way you choose. Data type: String
ReverseCategory	Specifies whether to reverse the sequence of the category field value. Data type: Boolean
ReverseSeries	Specifies whether to reverse the sequence of the series field value. Data type: Boolean
RowIndex	Specifies the Y coordinate of the object relative to its parent container when exported to Excel or CSV, measured in cells. The Columned property at the report tab level must be set to true for this property to take effect. Data type: Integer
RowNumber	Specifies the number of rows which will be the object's height in the exported Excel or CSV file. The Columned property at the report tab level must be set to true for this property to take effect. Data type: String
SecondaryDataFilter	Specifies the data format for the secondary value axis (the Y2 axis) to display the tick mark labels in the way you choose. Data type: String
SeriesEndOffset	The four properties work together to control the range of the data being displayed on a chart: CategoryStartOffset, CategoryEndOffset, SeriesStartOffset, and SeriesEndOffset. SeriesEndOffset specifies the ending offset of the series. Data type: Integer
SeriesFilter	Specifies the data format for the series axis (the Z axis) to display the tick mark labels in the way you choose. Data type: String
SeriesStartOffset	The four properties work together to control the range of the data being displayed on a chart: CategoryStartOffset, CategoryEndOffset, SeriesStartOffset, and SeriesEndOffset. SeriesStartOffset specifies the starting offset of the series. Data type: Integer
SeriesValueEncoding	Specifies the encoding format for values on the series axis. Data type: String
ShowLegend	Specifies whether to make the legend visible in a chart. Data type: Boolean
SortCategory	Specifies the sorting order for the category field values. Data type: Enumeration
SortSeries	Specifies the sorting order for the series field values. Data type: Enumeration
Style	The property can be used in two ways. • Specifies a style to be applied to the object. This works when a StyleGroup has been specified at the report sheet level and when there are styles in the StyleGroup that can be applied to the object. • Specifies a CSS selector to be applied to the object. Data type: String
Suppressed	Specifies whether to show the object in the design area and in the report result. All formulas and calculations will be skipped if the property is set to true. Data type: Boolean **Note:** When both the Invisible and Suppressed properties of an object are set to true, Suppressed has the higher priority.
SuppressedIfNoRecords	Specifies whether to display the object in the report result when no record is returned to its parent data container. Data type: Boolean
SwapGroup	Specifies to display values from different data fields by switching data between the category and series axes, the category and value axes of a chart. By default, this property is set to false, which means no switch will take place in the chart. • **Switching between the category and series axes** If you create a chart by adding fields to all of its axes, and set SwapGroup to true, data on the category and series axes will be switched. • **Switching between the category and value axes** If you create a chart by adding one field to the category axis and several to the value axis, and set SwapGroup to true, data on the category and value axes will be switched. Data type: Boolean **Note:** If the chart has only one field on the category axis and one on the value axis, even though the property value is set to true, there will be no switch take place on the two axes.
TOCAnchor	Specifies whether to include the object in the TOC Browser for the report. Data type: Boolean
Width	Specifies the width of the object, in inches or centimeters. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. Data type: Float
XHyperLink	Specifies whether to add a hyperlink that refers to another report or a website to the X axis labels. Data type: String
XHyperLinkTarget	Specifies the target window or frame in which to display the content the XHyperlink property specifies. • **<Server Setting>** Loads the linked file according to setting of the Pop Up New Window for Links option in the Customize Profile panel of the Profile dialog in JReport Server. • **New Window** Loads the linked file into a new window. The window is not named. • **Whole Window** Loads the linked file into the full browser window. • **Same Frame** Loads the linked file into the same frame as the link. • **Parent Frame** Loads the linked file into the parent frame of the frame that contains the link. • **Other Frame** Loads the linked file into some other specified frame. If the frame name does not exist, the linked file will be loaded into a new window. Data type: String
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. Data type: Float
ZHyperLink	Specifies whether to add a hyperlink that refers to another report or a website to the Z axis labels. Data type: String
ZHyperLinkTarget	Specifies the target window or frame in which to display the content the ZHyperlink property specifies. If the specified frame name does not exist, the linked document will be loaded into a new window. • **<Server Setting>** Loads the linked file according to setting of the Pop Up New Window for Links option in the Customize Profile panel of the Profile dialog in JReport Server. • **New Window** Loads the linked file into a new window. The window is not named. • **Whole Window** Loads the linked file into the full browser window. • **Same Frame** Loads the linked file into the same frame as the link. • **Parent Frame** Loads the linked file into the parent frame of the frame that contains the link. • **Other Frame** Loads the linked file into some other specified frame. If the frame name does not exist, the linked file will be loaded into a new window. Data type: String

Getting report metadata information

Dataset information can be got by the method getDataSet(), which returns the dataset object. Then the following report metadata information can be got from the dataset:

- Basic information
- Query information
- Business view information
- Business/Report cube information (Deprecated)
- Data buffer information

Getting basic information

- **getName()**
  Returns the dataset name.
- **getDataSourceName()**
  Returns the name of the data source used to build the data component.
- **getResourceType()**
  Returns the resource type. 1 means query, 2 means business view, and 3 means business/report cube.
- **getBlName()**
  Returns the information of the query, imported SQL, stored procedure, user defined data source, business view, or business/report cube used to build the data component.
- **getMaxRecords()**
  Returns the maximum number of records you want to display for all of the data containers using this dataset.
- **setMaxRecords(int maxRecords)**
  Sets the maximum number of records.
- **getDataDriver()**
  Returns the path of the cached query result file that is used as the data resource.
- **setDataDriver(String dataDriver)**
  Sets the path of the cached query result file.
- **getSecurityPolicyName()**
  Returns the security policy name that is applied to the data component.
- **setSecurityPolicyName(String securityPolicyName)**
  Sets the security policy name.

Getting query information

The following methods are used to get information of the query, imported SQL or other data objects that are used to create the dataset.

- **getQueriableName()**
  Returns the name of the query, imported SQL, stored procedure, or user defined data source.
- **getDataFieldNames()**
  Returns a collection of data field mapping names that the dataset uses.

Getting business view information

The following methods are used to get the business view information that is used to create the dataset.

- **getBVName()**
  Returns the business view name.
- **getQueriableName()**
  Returns the name of the query, imported SQL, stored procedure, or user defined data source that is used to create the business view.
- **getElementNames()**
  Returns a collection of the qualified names of the business view elements that the dataset uses.
- **getDisplayName(String qualifiedName)**
  Returns the display name of the business view element that has the qualified name.
- **getMappingName(String qualifiedName)**
  Returns the mapping name of the queriable object data field used to create business view element that has the qualified name.

Getting business/report cube information (Deprecated)

The following methods are used to get the business/report cube information that is used to create the dataset.

- **getCubeName()**
  Returns the business/report cube name.
- **getQueriableName()**
  Returns the name of the query, imported SQL, stored procedure, or user defined data source that is used to create the business/report cube.
- **getElementNames()**
  Returns a collection of the qualified names of the business/report cube elements that the dataset uses.
- **getDisplayName(String qualifiedName)**
  Returns the display name of the business/report cube element that has the qualified name.
- **getMappingName(String qualifiedName)**
  Returns the mapping name of the queriable object data field used to create business/report cube element that has the qualified name.

Getting data buffer information

- **getMaxPageNumber()**
  Returns the maximum number of pages in the data buffer.
- **setMaxPageNumber(int maxPageNumber)**
  Sets the maximum number of pages.
- **getRecordsPerPage()**
  Returns the number of records in each page in the data buffer.
- **setRecordsPerPage(int recordsPerPage)**
  Sets the number of records in each page.

•

Getting a Catalog API instance

The method public CatalogAPI getCatalogAPI() is used for getting a Catalog API instance.

•

Getting objects and object information

- **getDBFields(String handle)**
  Gets DBFields that can be used in the report and returns a mapping name array.
- **getControlFields(String handle, String name)**
  Gets fields that control property values at runtime.
- **getFormulae(String handle)**
  Gets formulas that can be used in the report and returns a mapping name array.
- **getSummaries(String handle)**
  Gets summaries that can be used in the report and returns a mapping name array.
- **getParameter()**
  Gets parameters in the catalog and returns a mapping name array.
- **getQueries()**
  Gets queries in the catalog and returns a mapping name array.
- **getQueryName(String handle)**
  Gets the query name of a report.
- **getProcedures()**
  Gets stored procedures in the catalog and returns a mapping name array.
- **getSections(String handle, boolean visible), getSections(String handle)**
  Gets handles of sections in a report.
- **getGroups(String handle)**
  Gets handles of groups in a report.
- **public GroupInfo[] getGroupInfo(String handle)**
  Gets the group information in a report.
- **public ReportSortInfo getReportSortInfo(String handle)**
  Gets the sort information of a report.
- **public SortInfo[] getSortInfo(java.lang.String handle)**
  Gets the sort information of a group.
- **getSQLs()**
  Gets the SQLs in the catalog.
- **getUDS()**
  Gets the UDSs in the catalog.
- **getViews()**
  Gets the VIEWs in the catalog.
- **getObjectType(String handle)**
  Gets the type information of an object.
- **getObjectInfo(String handle)**
  Gets the handle of an object in a report.

Parameters

- handle - Report node handle.
- name - The property name.
- visible - A Boolean type parameter which specifies that the section is visible.

Getting the instance name and class type of an object

- **getInstanceName(String handle)**
  Gets the instance name of an object, and returns the instance name. If it fails, it will return null.

- **getClassType(String handle)**
  Gets the class type of an object and returns one of the class type values that are defined in the Designer class. If an error or a warning occurs, it will return UNKNOWN.

Parameter

- handle - Object handle.

•

Moving objects

The method public boolean changeZOrder(String objectHandle, int zorderType) is used to move an object to front/back, or forward/backward.

Parameters

- objectHandle - Handle of the object whose z-order will be moved.
- zorderType - Direction to which the object will be moved.

•

Getting editing information

- **getWarning()**
  Gets and returns the warning message. If the operation is successful, it will return null.
- **clearWarning()**
  Clears the warning message.
- **getError()**
  Gets and returns the error message. If the operation is successful, it will return null.
- **clearError()**
  Clears the error message.
- **clearMsg()**
  Clears all warning and error messages.
- **setLog(OutputStream log, String encoding)**
  Sets the log to get error, debug and other information.
- **writeLog(String msg)**
  Writes all messages to the log.
- **closeLog() throws IOException**
  Closes the log.

•

Exiting from the editing status

- **exit()**
  Exits from the editing status, releases all the resources and saves all reports and the catalog.

- **quit()**
  Quits the editing status without saving and releases all resources. It will return false if it fails, otherwise true.

•

Concurrency control

Most of the methods in JReport Design API can also be used when programming with JReport Server Design API. There are however, some methods which are unique to JReport Server Design API. When programming with it, you should use these methods by using the parameters listed in MultiUserCatalogAPI and MultiUserDesigner, which can be found in the JReport Javadoc stored in `<install_root>\help\api`. The methods have parameter UID, and they are used to access private catalog resources while another user is editing the catalog in MultiUserCatalogAPI with the same UID; however, all the standard Catalog and Design API methods without the UID are still supported.

Notes:

- The number of concurrent reports and users will depend on your license.
- A user does not mean a person but means a session. Concurrent users mean concurrent sessions.
- When using JReport Server Design API, you must make sure that each user can only edit one report and that each report can only be edited by one user at the same time.
- MultiUserDesigner allows multiple users to edit a report; however, it only supports report sets with a single tab as designed in V7 or earlier JReport Designer. It can not be used with report sets that contain multiple tabs.
- For details about JReport Server Design API special methods, see the JReport Javadoc in `<install_root>\help\api`.

Illustrative examples

JReport provides you with some sample programs which demonstrate how to create and edit reports using the Design API in `<insatll_root>\help\samples\APIDesign`. They are:

- **TestDesignEditInvoice.java**
  An example of how to modify an existing page report template.
- **TestDesignGraph.java**
  An example of designing a report with a chart.
- **TestDesignInvoice.java**
  An example of designing a report containing a banded object.
- **TestDesignSubreport.java**
  An example of designing a report with a subreport.
- **TestInsertCrossTabIntoBanded.java**
  An example of inserting a crosstab into a banded object.
- **TestInsertCrossTabIntoReport.java**
  An example of designing a report with a crosstab.
- **TestMutiUserDesign.java**
  An example of creating two reports at the same time.

The following are three specific examples of using the Design API:

- Example 1: TestDesignInvoice.java
- Example 2: TestMutiUserDesign.java
- Example 3: TestDesignEditInvoice.java

Example 1: TestDesignInvoice.java

TestDesignInvoice.java in `<insatll_root>\help\samples\APIDesign` is primarily designed for creating a report with the methods provided by the Design API. Here is an explanation of how TestDesignInvoice.java works.

- **Creating an instance of Designer**
  ```
  desg = new Designer(path, cat);
  ```
  The constructor of Designer has two parameters: path and name. They are respectively the path and name of the catalog with which the report will be created.

- **Setting log if necessary**
  ```
  desg.setLog(new FileOutputStream(log), "8859_1");
  ```
  The output stream log points to the path and file in which to write log messages. Parameter "8859_1" is the encoding for the messages.

- **Getting handles**
  ```
  handles = desg.getHandles(report, Designer.PAGE);
  ```
  This method gets all handles of PAGE type and returns a handle array. Parameter report is the parent handle, and Designer.PAGE is a class type value which is defined in the Designer class.

- **Creating a report set**
  ```
  // Create a report set. Use the name from the -name parameter.
  reportset = designer.createReportSet(name);
  ```
  This method creates a new report and returns its handle. Parameter name is the name of the report that is to be created.

- **Adding a connection**
  ```
  // Register the existing query "WorldWideSales" to this report set.
  dataset = designer.addDataset(reportset, "Data Source 1",
  "WorldWideSales","dsWorldWideSales", "",Designer.DATASET_TYPE_QUERY);
  ```
  This method adds a dataset to the report. The dataset must exist in the referenced catalog.

- **Adding a report tab**
  ```
  // Add a report tab named "Invoice" to the report set.
  report = designer.addReport(reportset, "Invoice");
  ```
  This method creates a new report and returns its handle. Parameter name is the name of the report that is to be created.

- **Adding a banded object**
  ```
  // Insert the banded object for the invoice report
  banded = designer.insertBandedObject(reportbody, true, true, true);
  ```
  This method adds a banded object into our report tab.

- **Inserting a group panel**
  ```
  gGroupInfo ginfo =new GroupInfo( "Customers_Customer ID", GroupInfo.DESCENDING );
  groupCust = designer.insertSectionGroup( handles[0], ginfo );
  ```
  This method inserts a new group panel and returns its handle. Parameter handles[0] is the parent node handle. The newly inserted object will be in this node. Customers_Customer ID is the mapping name to group by.

- **Inserting a label**
  ```
  handle = desg.insert(header, Designer.LABEL, "Label1");
  ```
  This method inserts a label named label1 into the group header. Parameter header is the parent handle, the newly inserted object will be in this node. desg.LABEL is the class type that you want to insert, and Label1 is the object name to be inserted.

- **Inserting DBField**
  ```
  handle = desg.insert(header, Designer.DBFIELD, "DBField1", "Orders_Order ID");
  ```
  This method inserts a DBField named DBField1 into the group header. Parameter header is handle of the parent node; Designer.DBFIELD is class type that is to be inserted, and Orders_Orders ID is the mapping name of the database field.

- **Inserting a formula, parameter and summary**
  To insert a formula, parameter or summary, use the same method as for inserting a DBField. Note that the formula, parameter or summary that is to be inserted should already exist in the catalog.

  For example, in the sample program, it is made sure that the parameter PToday, formula customeraddress12, and summary Sum_total9 already exist in the catalog by the following:

  ```
  handle = desg.insert(header, Designer.PARAMETER, "Parameter1", "PToday");
  handle = desg.insert(header, Designer.FORMULA, "formula1", "customeraddress12");
  handle = desg.insert(footer, Designer.SUMMARY, "summary1", "Sum_total9");
  ```

- **Inserting an image, box and line**
  To insert an image, box or line, use the same method as for inserting a label.

  For example, in the sample program, you can insert the image, box and line as follows:

  ```
  handle = desg.insert(header, Designer.IMAGE, "Image1");
  handle = desg.insert(report, Designer.BOX, "Box1");
  handle = desg.insert(report, Designer.LINE, "Line1");
  ```

- **Setting the property value**
  To set the property value, you should provide three parameters: object handle, property name and property value.

  In the sample program, you can set the property value as follows:

  ```
  desg.set(handle, "Width", 1.74f);desg.set(handle, "Height", 0.35f);desg.set(handle,
  "Text", "Invoice");desg.set(handle, "Alignment", desg.LEFTCENTER);desg.set(handle,
  "Foreground", new Color(0x900000));desg.set(handle, "Bold", true);desg.set(handle,
  "FontSize", 0.31f);
  ```

- **Exiting and closing all reports**
  ```
  desg.exit();
  ```
  This method ends the editing status, releases all resources and saves all of the reports and the catalog.

- **Compiling and running the sample program**
  To compile and run the sample program, you should add report.jar, sac.jar, log4j-1.2.8.jar and JREngine.jar with their path into the class path.

  For example, you can use the following command to compile the sample program:

  ```
  C:\JReport\Designer\help\samples\APIDesign>javac -classpath "C:\JReport\Designer\lib
  \report.jar;C:\JReport\Designer\lib\sac.jar;C:\JReport\Designer\lib\JREngine.jar;C:
  \JReport\Designer\lib\log4j-1.2.8.jar" TestDesignInvoice.java
  ```

  When you run the sample program, you should provide two or three parameters. If you want to use two parameters, they should be catalog path and catalog name, and the command line should be:

  ```
  C:\JReport\Designer\help\samples\APIDesign>java -Dreporthome="C:\JReport\Designer" -
  classpath "C:\JReport\Designer\lib\jrengine.jar;C:\JReport\Designer\lib\sac.jar;C:
  \JReport\Designer\lib\resource_en_US.jar;C:\JReport\Designer\lib\report.jar;C:\JReport
  \Designer\lib\log4j-1.2.8.jar;C:\JReport\Designer\lib\hsqldb.jar" TestDesignInvoice -
  path=C:\JReport\Designer\Demo\Reports\TutorialReports -catalog=JinfonetGourmetJava.
  cat -report=TestInvoice.cls
  ```

  If you want to use three parameters, they should be catalog path, catalog name and log file with a full path name, and the command line should be:

  ```
  C:\JReport\Designer\help\samples\APIDesign>java -Dreporthome="C:\JReport\Designer" -
  classpath "C:\JReport\Designer\lib\JREngine.jar;C:\JReport\Designer\lib\sac.jar;C:
  \JReport\Designer\lib\resource_en_US.jar;C:\JReport\Designer\lib\report.jar;C:\JReport
  \Designer\lib\log4j-1.2.8.jar;C:\JReport\Designer\lib\hsqldb.jar" TestDesignInvoice -
  path=C:\JReport\Designer\Demo\Reports\TutorialReports -catalog=JinfonetGourmetJava.
  cat -report=TestInvoice.cls -log=C:\JReport\Designer\logs\designer.log
  ```

  Here, it is assumed that JReport Designer has been installed to `C:\JReport\Designer`, and the current directory when you execute these commands is `C:\JReport\Designer\help\samples\APIDesign` (location of the sample programs). After running the program, you will find a newly created file TestInvoice.cls in `C:\JReport\Designer\Demo\Reports\TutorialReports`.

Example 2: TestMutiUserDesign.java

TestMutiUserDesign.java in `<insatll_root>\help\samples\APIDesign` is primarily designed for creating two reports at the same time using the methods provided by the Design API. The MultiUserDesign API can only create page reports in the V7 and earlier format. These run in later versions of JReport but can not include multiple page tabs, each template can only contain a single report tab.

With this sample program, two reports are created:

- **Creating the first instance of MultiUserDesigner**
  ```
  desg = new MultiUserDesigner(path, cat, userInfo);
  ```

  The constructor of MultiUserDesigner has three parameters: path, name and userInfo. They are respectively the path and name of the catalog with which the reports will be created, and the user ID and license key which are provided by JReport.

- **Creating the first report**
  ```
  report = desg.create(name, "invoice", User)
  ```

  This method creates a new report and returns its handle. Parameter name is the name of the report that is to be created, Invoice is the name of query in the catalog file SampleReports.cat on which the report is created, and User is the user who creates the report.

- **Getting handles**
  ```
  handles = desg.getHandles(report, Designer.PAGE);
  ```

  This method gets all handles of PAGE type and returns a handle array. Parameter report is the parent handle, and Designer.PAGE is the class type value defined in the Designer class.

- **Inserting a parameter, Dbfield, formula**
  To insert a parameter, Dbfield or Formula. The following methods are examples of the special methods mentioned before. User is the name of each concurrent user.

  ```
  handle = desg.insert(User, header, Designer.PARAMETER, "Parameter1", "PToday")
  handle = desg.insert(User, header, Designer.DBFIELD, "DBField1", "Orders_Order ID")
  handle = desg.insert(User, header, Designer.FORMULA, "formula1", "customeraddress12")
  ```

- **Creating the other report**
  User is the name of each concurrent user. There are two users in this demo.

  ```
  DesignerReport DESInvoice = new DesignerReport("Myinvoice", test.desg, "designer1");
  DesignerReport DESregionsalesgraph = new DesignerReport("Myregionsalesgraph", test.
  desg, "designer3");
  ```

Example 3: TestDesignEditInvoice.java

TestDesignEditInvoice.java in `<insatll_root>\help\samples\APIDesign` shows how to modify a report using the methods of the Design API. It demonstrates how to change the image in the report TestInvoice.cls.

Here is an explanation of how TestDesignEditInvoice.java works with the methods provided by the Design API.

- **Creating an instance**
  ```
  desg = new Designer(path, cat);
  ```
  The constructor of Designer has two parameters: path and name. They are respectively path and name of the catalog with which the report will be created.

- **Setting log**
  ```
  desg.setLog(new FileOutputStream(log), "8859_1");
  ```
  The output stream log points the path and file in which to write log messages. Parameter "8859_1" is the encoding for the messages.

- **Opening an existing report**
  ```
  report = desg.open(name);
  ```
  This method opens an existing report corresponding to the name specified, and returns the handle of the report.

- **Getting handles**
  ```
  handle = desg.getHandles(report, Designer.IMAGE, -1)[0];
  ```
  This allows the handle of the object to be modified.

- **Changing the property value**
  ```
  // change the image "coffee.gif" to "CoffeeLogo.gif"
  desg.set(handle, "PictureName", "CoffeeLogo.gif");
  // change the image position
  desg.set(handle, "X", 4.67f);
  // change the image width and height
  desg.set(handle, "Width", 1.70f);
  desg.set(handle, "Height", 0.81f);
  ```

- **Exiting and closing all reports**
  ```
  desg.exit();
  ```

- **Compiling and running the sample program**
  To compile and run the sample program, you should add report.jar, sac.jar, log4j-1.2.8.jar and JREngine.jar with their path into the class path.

  For example, you can use the following command to compile the sample program:

  ```
  C:\JReport\Designer\help\samples\APIDesign>javac -classpath "C:\JReport\Designer\lib
  \report.jar;C:\JReport\Designer\lib\sac.jar;C:\JReport\Designer\lib\JREngine.jar;C:
  \JReport\Designer\lib\log4j-1.2.8.jar" TestDesignEditInvoice.java
  ```

  When you run the sample programs, you should provide two or three parameters. If you want to use two parameters, they should be catalog path and catalog name. The command line should be:

  ```
  C:\JReport\Designer\help\samples\APIDesign>java -Dreporthome="C:\JReport\Designer" -
  classpath "C:\JReport\Designer\lib\JREngine.jar;C:\JReport\Designer\lib\sac.jar;C:
  \JReport\Designer\lib\report.jar;C:\JReport\Designer\lib\log4j-1.2.8.jar;C:\JReport
  \Designer\lib\hsqldb.jar;" TestDesignEditInvoice -catalog=JinfonetGourmetJava.cat -
  report=TestInvoice.cls
  ```

  If you want to use three parameters, they should be catalog path, catalog name and log file with a full path name. The command line should be:

  ```
  C:\JReport\Designer\help\samples\APIDesign>java -Dreporthome="C:\JReport\Designer" -
  classpath "C:\JReport\Designer\lib\JREngine.jar;C:\JReport\Designer\lib\sac.jar;C:
  \JReport\Designer\lib\report.jar;C:\JReport\Designer\lib\log4j-1.2.8.jar;C:\JReport
  \Designer\lib\hsqldb.jar;" TestDesignEditInvoice -path=C:\JReport\Designer\Demo
  \Reports\TutorialReports -catalog=JinfonetGourmetJava.cat -report=TestInvoice.cls -
  log=C:\JReport\Designer\logs\design.log
  ```

  Here, it is assumed that JReport Designer has been installed to `C:\JReport\Designer`, and that the directory that is current when you execute these commands is `C:\JReport\Designer\help\samples\APiDesign` (location of the sample programs). After running the program, you will find a modified file TestInvoice.cls in `C:\JReport\Designer\Demo\Reports\TutorialReports`.

JReport Result Viewer Bean

JReport Result Viewer Bean enables you to:

- Show the report result files (.rst extension).
- Export the report result files to multiple file formats (HTML, PDF, XLS, RTF, Postscript, Text, CSV, Applet, E-mail) and to printer.

Reference: See also the JReport Javadoc jet.bean.JRResultViewer class in `<install_root>\help\api`.

The following sections provide a detailed introduction about JReport Result Viewer Bean:

- Installing the Result Viewer Bean
- Result Viewer Bean properties
- Programming with the Result Viewer Bean

Installing the Result Viewer Bean

When you install JReport Designer, which by default is located in `C:\JReport\Designer`, JReport Result Viewer Bean is also installed at the same time. After installation, you will have the main components in `<install_root>` as follows:

- **lib\JRResultViewer.jar***
  Jar file for the Result Viewer Bean.
- **lib\log4j-1.2.8.jar***
  Jar file used for the JReport Engine logging system.
- **lib\activation.jar**
  Jar file for exporting to mail.
- **lib\mail.jar**
  Jar file for exporting to mail.
- **lib\resource_en_US.jar**
  Archive file used when exporting to XML.
- **lib\xml-apis.jar, xercesImpl.jar**
  Jar files used when exporting to XML.
- **lib\itext_1.5.4.jar**
  Jar file used when exporting to PDF.
- **lib\iTextAsian.jar**
  Jar file used for the Asian fonts.
- **lib\poiHSSF_151.jar**
  Jar file used when exporting to Excel.
- **SwingDesigner.jar**
  If RPTExecutor UDO has been inserted into a report, to view the correct result, this jar file should be added to the class path that starts the Result Viewer Bean.
- **Readme.htm**, **License.htm**
  Release information.
- **help\samples\APIViewResult\TestResultBean.java**
  A sample program to demonstrate how to display the result.

Notes:

- * means that these files are required when you call the Result Viewer Bean.
- Other files without * should be added to your class path to call the Result Viewer Bean.

  For example, when you want to use the Result Viewer Bean to export a result file to mail, you should add mail.jar and activation.jar with a valid path to your class path, and when you want to export the result file to XML, you should add resource_en_US.jar with a valid path to your class path.

Result Viewer Bean properties

JReport Result Viewer Bean has four properties: reportName, showInfoLevel, logFile and loadReport. The following methods are used to set the values of these properties:

- setReportName ("ReportName")
- setTempPath("theTemporaryPath");
- setShowInfoLevel (vDebug/vError/vOff)
- setLogFile ("LogFileName")
- setLoadReport ("Yes/No")

Parameters

- ReportName - Specifies the name of the JReport result file that is to be viewed. It contains both the path information and the file name.

  > **Note:** Here the report name is the result file name with an .rst extension (NOT the report file .cls).

- theTemporaryPath - The temporary file path. When the bean runs, the temporary files will be written in this path. You should set the full path of the temporary files in this method as:

  ```
  bean.setTempPath("c:\\temp");
  ```

  If you do not set the temporary path, the default path is the directory from where the Bean will be run.

- vDebug/vError/vOff - These are used for showing and turning off debug and error information. LogFileName specifies where the debug, error and other information will go. LogFileName contains both the path and the file name. If no path or file is specified, the default will be used.

- Yes/No - Indicates whether or not to start to view the result of the report (the file specified by ReportName).

Programming with the Result Viewer Bean

JReport Result Viewer can be used to view local JReport result files generated by JReport Engine. You can show the report result file (.rst) and export the report result to multiple file formats and to printer, by calling JReport Result Viewer Bean in three different ways: by JReport Designer, by JReport Server, or by JReport Engine Bean.

The following topics describe how to program with JReport Result Viewer Bean:

- Preparations
- Using the Result Viewer Bean
- An illustrative example

Preparations

Before programming with the Result Viewer Bean, you need to import the classes of the Result Viewer Bean by adding the following import statements to the beginning of your program. The first import statement is added to import the bean class:

```
import jet.bean.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
```

Then, you need to create an instance.

To run with the JReport Result Viewer Bean, you should first create an instance of it. The constructor of the Result Viewer Bean has no parameter. After the object has been created, you should then set the report name and temp path (if not set, the current path will be used).

```
bean = new JRResultViewer();
bean.setShowInfoLevel(bean.vDebug | bean.vError);
bean.setReportName("c:\\test\\customerlist.rst");
bean.runReport("yes");
```

With the JReport Result Viewer Bean object, you can show the report result and export it to other formats or to printer. You do not need to create a bean for each report. The Result Viewer Bean provides many useful methods.

Using the Result Viewer Bean

After making the required preparations, you can now use the Result Viewer Bean to perform tasks.

The following topics detail how to run with the Result Viewer Bean to achieve certain goal:

- Exporting report result to different file formats
- Printing reports
- Getting the page orientation of a report page
- Browsing report pages
- Zooming reports
- Getting field results by listening to the click event
- Setting the report communicator
- Setting the user input stream
- Exiting from the Result Viewer Bean

Exporting report result to different file formats

There are several methods with which JReport Result Bean can export a result file (.rst extension) to different file formats (.htm, .pdf, .txt, and so on):

- **Export to HTML**
  - exportToHtml(string HTMLFileName)
  - exportToHtml(string HTMLFileName, HtmlParameter Parameter)
  - exportToHtml(string HTMLFileName, boolean bChartApplet, boolean isMultiFile, boolean bUsingTable, boolean bHyperlink, boolean bPageNum, boolean bAbsolute, int iBrowser)
  - exportToHtml(string HTMLFileName, boolean bChartApplet, boolean isMultiFile, boolean bUsingTable, boolean bHyperlink, boolean bPageNum, boolean bAbsolute, int iBrowser, Drilldownfile)
  - exportToHtml(string RSTFileName, string HTMLFileName, boolean bChartApplet, boolean isMultiFile, boolean UsingTable, boolean Hyperlink, boolean PageNum, boolean bAbsolute, int iBrowser, boolean bincdrilldownfile, boolean noMargin)
  - exportToHtml(string rstFileName, string HTMLFileName, boolean bChartApplet, boolean isMultiFile, boolean bUsingTable, boolean bHyperlink, boolean bPageNum, boolean bAbsolute, int iBrowser, boolean bincdrilldownfile, boolean noMargin, int resolution)
- **Export to Page Report Result**
  - exportToDhtml(string HTMLFileName)
  - exportToDhtml(string RSTFileName, string HTMLFileName, boolean bChartApplet, boolean isMultiFile, boolean bUsingTable, boolean bHyperlink, boolean bPageNum, boolean bAbsolute, int iBrowser, boolean bincdrilldownfile, boolean noMargin)
  - exportToDhtml(string HTMLFileName, boolean bChartApplet, boolean isMultiFile, boolean bUsingTable, boolean bHyperlink, boolean bPageNum, boolean bAbsolute, int iBrowser, boolean bincdrilldownfile, boolean noMargin, int resolution)
  - exportToDhtml(string HTMLFileName, boolean bChartApplet, boolean isMultiFile, boolean bUsingTable, boolean bHyperlink, boolean bPageNum, boolean bAbsolute, int iBrowser)
  - exportToDhtml(string HTMLFileName, boolean bChartApplet, boolean isMultiFile, boolean bUsingTable, boolean bHyperlink, boolean bPageNum, boolean bAbsolute, int iBrowser, boolean bincdrilldownfile)
  - exportToDhtml(string HTMLFileName, boolean bChartApplet, boolean isMultiFile, boolean bUsingTable, boolean bHyperlink, boolean bPageNum, boolean bAbsolute, int iBrowser, boolean bincdrilldownfile, boolean noMargin)
- **Export to PDF**
  - exportToPdf(string PDFFileName)
  - exportToPdf(string FileName, PDFParameter Parameter)
  - exportToPdf(string PDFFileName, boolean noMargin)
  - exportToPdf(string PDFFileName, boolean noMargin, boolean isSimPrintMode)
  - exportToPdf(string PDFFileName, boolean noMargin, boolean isSimPrintMode, int isCompressImage)
  - exportToPdf(string RSTFileName, string PDFFileName, boolean noMargin, boolean isPrintMode, int isCompressImage, boolean isTransparent)
  - exportToPdf(string PDFFileName, boolean noMargin, boolean isSimPrintMode, int isCompressImage, boolean isTransparent, boolean bToc, boolean bDrill, string password)
  - exportToPdf(string PDFFileName, boolean noMargin, boolean isSimPrintMode, int isCompressImage, boolean isTransparent, boolean bToc, boolean bDrill)
- **Export to Text**
  - exportToText(string TextFileName, boolean isNormalText)
  - exportToText(string TextFileName, TextParameter Parameter)
  - exportToText(string TextFileName, boolean isNormalText, int udchWidth, int unchHeight)
  - exportToText(string TextFileName, boolean isNormalText, boolean isRepeat, char delimiter)
  - exportToText(string TextFileName, boolean isNormalText, boolean isRepeat, char delimiter, boolean compress)
  - exportToText(string TextFileName, boolean isNormalText, boolean isRepeat, char delimiter, boolean compress, boolean quoteMark)
  - exportToText(string TextFileName, boolean isNormalText, boolean isRepeat, char delimiter, int udchWidth, int udchHeight, boolean compress, int resolution, boolean quoteMark, boolean hasHeadfoot)
  - exportToText(string TextFileName, boolean isNormalText, boolean isRepeat, char delimiter, int udchWidth, int udchHeight, boolean compress, int resolution, boolean quoteMark, boolean hasHeadfoot, boolean forWindows)
  - exportToText(string TextFileName, boolean isNormalText, int udchWidth, int udchHeight, boolean compress, int resolution)
  - exportToText(string TextFileName, boolean isNormalText, int udchWidth, int udchHeight, boolean compress, int resolution, boolean quoteMark)
- **Export to Excel**
  - exportToXls(string XLSFileName)
  - exportToXls(string XLSFileName, ExcelParameter Parameter)
  - exportToXls(string XLSFileName, int isWordWrap)
  - exportToXls(string XLSFileName, int isWordWrap, boolean isBiff8)
  - exportToXls(string XLSFileName, int isWordWrap, boolean isBiff8, int param)
- **Export to PS**
  - exportToPS(string PSFileName)
  - exportToPS(string PSFileName, boolean noMargin)
  - exportToPS(string PSFileName, PSParameter Parameter)
- **Export to RTF**
  - exportToRtf(string RTFFileName)
  - exportToRtf(string RTFFileName, boolean noMargin)
  - exportToRtf(string RTFFileName, RTFParameter Parameter)
- **Export to XML**
  - exportToXML(string XMLFileName, boolean isMultiFile, boolean isOnlyData)
  - exportToXML(string XMLFileName, boolean isDataOnly, string xsdFile)
  - exportToXML(string XMLFileName, boolean isDataOnly)

Parameters

- HTMLFileName - The name of the HTML format result file with a full path.
- bChartApplet - A boolean value which specifies whether or not the chart in an HTML file is an applet. The default value is true.
- isMultiFile - A boolean value which specifies whether or not HTML is generated to a single file or to multiple files. The default value is true.
- bUsingTable - A boolean value which specifies whether or not to use HTML table format when exporting HTML files. The default value is false.
- bHyperLink - A boolean value which specifies whether or not a hyperlink is generated. The default value is true.
- bPageNum - A boolean value which specifies whether or not to generate page numbers. The default value is true.
- bAbsolute - A boolean value which specifies whether the font size is absolute or relative. The default value is true.
- iBrowser - An int value which specifies whether or not the web browser is IE. 0 means IE. The default value is 0.
- Drilldownfile - A boolean value which specifies whether or not to generate the drilldown HTML files in the file list. The default value is false.
- RSTFileName - The name of the result file (.rst extension) that is to be converted to the required format.
- PDFFileName - The name of the PDF format result file with a full path.
- isSimPrintMode - A boolean value which specifies whether or not to use Simulated Printing Mode (true) or Standard Mode (false) (To PDF). By checking the Standard Mode, the exporting speed will be much improved. The default value is false.
- isCompressImage - An int value which specifies the compression ratio of the image in a report (To PDF). The values range from 1 to 100. The default value is 20.
- isTransparent - A boolean value which specifies whether or not to keep the background of the GIF image transparent.
- TextFileName - The name of the Text format result file with a full path. true/false is to set whether this text file is in normal Text format or standard Data format. The standard Data format text file is a text file where each row represents a single record in the report. It can be used as a text datasource for exchanging data with other applications.
- isNormalText - A boolean value which specifies whether or not the text file is in normal text format. When it is set to false, the output format is in standard data format. The default value is true.
- isRepeat - A boolean value which specifies whether or not the same contents are repeated. The default value is false.
- delimiter - A character used in SDF (Standard Data Format) format. The delimiter will only be used when the "isNormalText" parameter is set to false. The default delimiter is a blank space.
- compress - A boolean value which specifies whether or not to remove empty lines or redundant delimiters in the exported text file. The default value is false.
- XLSFileName - The name of the Excel format report result file with a full path.
- PSFileName - The name of the Postscript format result file with a full path.
- RTFFileName - The name of the Rich Text Format file with a full path.
- noMargin - A boolean value which specifies whether or not to generate a file without a page margin. The default value is false.
- chWidth - User defined char width used in plain text format. It would be ignored if the value is 0 or less than 0. When exporting to Text, the default value is -1.
- chHeight - User defined char height which is used in plain text format. It would be ignored if the value is 0 or less than 0. When exporting to Text, the default value is -1.
- XMLFileName - The XML file name.
- isMultiFile - A boolean value which specifies whether XML is generated to a single file or to multiple files.
- isOnlyData - A boolean value which specifies whether or not only the database column information is exported to an XML file. Otherwise, all the report details, including all the properties will be exported. The default value is false.
- parameter - A parameter contains some special settings for exporting to different file formats.
- resolution - An int value which is used to specify resolution for different exported file formats.
- password - A string value which is used to set the password when exporting to a PDF file that contains a password.
- bToc - A boolean value which specifies whether or not to generate TOC for the PDF file.
- bDrill - A boolean value which specifies whether or not to apply drill-down function to the exported PDF file.
- quoteMark - A boolean value specifies whether or not quote characters will be kept in standard format text file.
- hasHeadfoot - A boolean value specifies whether or not page header and footer will be contained when exporting to TEXT.
- forWindows - A boolean value specifies whether the CR-LF in Windows convention or the LF in Unix convertion will be used as the end-of-line character.
- udchWidth - An int value which is specified for each unit of the horizontal density between columns.
- udchHeight - An int value which is specified for each unit of the vertical density between columns.
- isWordWrap - An int value which specifies a word wrap setting option.
  - KEEPORIGIN - Keeps all the original settings of each component's Word Wrap property as specified in the report.
  - NOWORDWRAP - Disables the Word Wrap property for all components.
  - WORDWRAPALL - Enables the Word Wrap property for all components.
- isBiff8 - A boolean value specifies whether or not to export reports to a Data Format file, which means only the report data will be exported without format.
- param - An int value specifies whether or not to include drawing objects in the exported Excel file. If isBiff8 is set to true, drawing objects will not be included in the result file, because they cannot be exported to Data Format files, which means only the report data will be exported without format.
  - NOEXPORTSHAPE - Drawing objects will not be exported to Excel files.
  - EXPORTSHAPE - Drawing objects will be exported to Excel files.
- XSDFile - A string value indicates the schema file on which the exported XML file is based. If it is set to null, a new XSD file will be output.

Notes:

- Method exportToHtml(string HtmlFileName) will use the method exportToHtml(string HTMLFileName, boolean bChartApplet, boolean isMultiFile, boolean bUsingTable, boolean bHyperlink, boolean bPageNum, boolean bAbsolute, int iBrowser) using default parameter values, such as: exportToHtml ("HtmlFileName", false, true, false, true, true, true, JRResultViewer.IE).
- Method exportToText(string TextFileName, boolean isNormalText) will use the method exportToText (string TextFileName, boolean isNormalText, boolean isRepeat, char delimiter) using default parameter values such as: exportToText("TextFileName", isNormalText, false, ' ').

Printing reports

There are five methods provided for printing a report.

- printReport()
- printReport(Object printJob, Object pageFormat, boolean bInteractive, boolean bInBackground, boolean bUseJDK11)
- printReport(Object printJob, Object pageFormat, boolean bInteractive, boolean bInBackground, boolean bUseJDK11, boolean bSeparateLargePage)
- printReport(java.lang.Object job, boolean bInteractive, java.lang.Object pf, boolean bInBackground, JReportPrintListener jreportprintlistener, int useJDK, java.lang.String jobName, java.util.Vector attributeInfo, java.lang.String printer)
- getPrintStatus()

Parameters

- bInteractive - A Boolean value. If you set it to false, the Print Setup dialog will not be shown.
- bInBackground - A Boolean value. If you set it to true, the Print dialog will not be shown.
- bUseJDK11 - A Boolean value. If you set it to true, it will use JDK1.1.x to print the report. Otherwise if you set it to false, it will use the JDK that you are running to print the report. The JDK 1.1 printing method is quick in speed but the printing quality cannot be guaranteed. The JDK 1.2 method will provide a satisfactory result, even for a .gif file, but it is slow in speed.
- bSeparateLargePage - A Boolean value. If the report page size is larger than the print paper size, and you want to separate a large report page during printing, you should set this property to true. If the print page size you choose is less than the report page size, the report will be printed in multiple pages serially. This means a large report will be separated automatically. For JDK1.2 or later, after you have set this parameter to true, the Page Format dialog will be shown. In this dialog, you can choose the page format in which you want to print. This page size will then be used for printing. For JDK1.1.x, however, you must first change the page format in the Print dialog (click the Property button, then choose the page format).
- job - The print job of this print action. It is an instance of PrintJob or PrinterJob.
- bUseJDK - An Integer value. If you set it to 0, it will use the method JDK1.1 to print the report. If you set it to 1, it will use the method JDK1.2 to print the report. If you set it to 2, it will use the method JDK1.4 to print the report. The JDK 1.1 printing method is quick in speed but the printing quality cannot be guaranteed. However, the JDK 1.2 method will return you with a satisfactory result, even for a gif file, but it is slow in speed. The method JDK1.4 provides further print options, such as specifying the paper tray.
- attributeInfo - A vector with String elements. Each string has been defined in interface PrintConfig which is located in package jet.bean. For example, you can implement PrintConfig and call its fields as shown below:

  ```
  attributeInfo.add(TRAY_MANUAL);
  attributeInfo.add(ORIENTATION_PORTRAIT).
  ```

Example 1

With JDK1.2 or later, if you use the JDK1.2 printing method (boolean bUseJDK11), you can call the printReport method as shown below:

```
PrinterJob printJob=PrinterJob.getPrinterJob();
bean.printReport(printJob, printJob.defaultPage(), false, false, false);
```

Or, if you do not want to use getPrinterJob, you can simply call:

```
bean.printReport(null, null, false, false, false);
```

Example 2

With JDK1.2 or later, if you use the JDK1.1 printing method, you are best recommended not to use getPrintJob, instead you can call:

```
bean.printReport(null, null, false, false, true);
```

Notes:

- The first method printReport() is called to start printing. It will take the third method using default parameter values such as:
  printReport(null, null, true, false, false, false)
- The second and third method allows you to pass the parameters - printJob, pageFormat, bInteractive, bInbackground, bUseJDK11 and bSeparateLargePage.
- The third printing method (with the bSeparateLargePage parameter) is not recommended. It may cause some unexpected errors, for example, you may get an extra blank page.
- After printing has started, the method getPrintStatus() can be used to get the print job status. Returned values of getPrintStatus() are:
  "No print job"
  "Busy"
  "OK"
  "No report load"
  Other error messages if there are errors.

Getting the page orientation of a report page

The method getPageOrientation() is used to get the orientation of the report page. You should call this method after calling the runReport method. The return value is an int value, and can be LANDSCAPE or PORTRAIT. If it is -1, it makes no sense, and is incorrect. For example, you may have called it before calling runReport method.

Browsing report pages

The methods listed here are also available to JReport Result Viewer Bean.

- The methods getCurrentPageNumber() and getTotalPageNumber() are used to return the current page number and the total number of pages of the current report onto the screen.
- The method int gotoPage(int pageNumber) is used to display the page specified by PageNumber on the screen. If PageNumber is out of the range of the current report, the first page (if the pageNumber is less than 0), or the last page will be shown on screen. The returned value is the current page number.
- The methods firstPage(), prevPage(), nextPage() and lastPage() are used to show the specified page on the screen.

The return value of the above methods is a string, and can be:

- "OK"
- "No more page"
- Other error messages such as "No report load".

Zooming reports

If you want to zoom in or out of a report, there is a method named zoom(ratio) that will meet this requirement.

Parameter

- ratio - Integer type, representing a percentage value.

Example 1

If you want to enlarge the report by two times the original size, you can use the method in this way:

```
zoom(200)
```

Example 2

If you want to reduce the report size by half, you can use the method in this way:

```
zoom(50)
```

Getting field results by listening to the click event

When you click a field in the viewer, you can get the entire record of the field. The interface in package jet.thinviewer named ClickActionListener can implement this function. This interface is defined as:

```
interface ClickActionListener
{
        void clickOnField(jet.connect.Record currentRecord);
}
```

If you have a class which wants to listen to the click event of a field, this class can implement this interface. Then, after you have instantiated a bean, you can add the ClickActionListener.

Note: This action must be implemented before running the report. Finally after the report has been shown and you click on a field, the ClickActionListener's method clickOnField will be called, and you will then get the values of all columns in that field from the parameter currentRecord.

Setting the report communicator

The method SetReportCommunicator(Communicator) is used to set the report communicator.

Parameter

- communicator - The Communicator here is a class defined by JReport. After users set the communicator, the viewer will load the report from this communicator, not the result file.

You are recommended to use getReportCommunicator() in JReport Engine Bean, and use the method setCommunicator() in JReport Result Viewer Bean. That is,

```
resultViewerBean.setCommunicator(engineBean.getReportCommunicator());
```

Setting the user input stream

To view the result file with JReport Result Viewer, you will usually set the report name to a physical .rst file, or if you want to set the stream of the .rst file, you can use the methods below:

- setInputStream(InputStream ins, boolean zip)

  Parameters

  - ins - The user input stream.
  - zip - Whether the result is compressed or not. To see whether the result is compressed or not, you can call the method below:

- boolean isZipStream(InputStream ins)

  Parameter

  - ins - The user input stream.

  Return

  - true - Compressed stream.
  - false - Uncompressed stream.

Exiting from the Result Viewer Bean

The method exit() should be called when you want to quit from JReport Result Viewer Bean. It will perform some clean up work, such as deleting temporary files. It is important to free up the resources of the result file, so remember to follow this method.

An illustrative example

There is a sample program called TestResultBean.java in `<install_root>\help\samples\APIViewResult` included in the JReport Server and JReport Designer installation.

The properties for the report are very simple, and you will only need to set the ReportName. Note that here the report name is the report result .rst file generated by JReport Engine. It is not a .cls file. By calling the runReport and show() methods, the report result will be displayed.

The report result can be exported to other formats such as HTML, PDF, RTF or to a printer. You will need to use the appropriate methods and parameters.

To run TestResultBean.java, you can use the following command to compile the sample program first (assume that JReport Designer has been installed to `C:\JReport\Designer`):

```
C:\JReport\Designer\help\samples\APIViewResult>javac -classpath "C:\JReport\Designer\lib
\JRResultviewer.jar;C:\JReport\Designer\lib\log4j-1.2.8.jar;C:\JReport\Designer\lib
\itext_1.5.4.jar" TestResultBean.java
```

JReport Catalog Bean

JReport Catalog Bean provides a programming API for manipulating and accessing a JReport catalog. A catalog is used for holding all the resources used for building reports in JReport.

JReport Catalog Bean enables you to:

- Create an empty catalog.
- Add objects to a catalog.
- Remove objects from a catalog.
- Get an object description in a catalog.

The following are some specific topics about JReport Catalog Bean:

- Installing the Catalog Bean
- Catalog Bean classes
- Programming with the Catalog Bean

Installing the Catalog Bean

When you install JReport Designer, which by default is located in `C:\JReport\Designer`, JReport Catalog Bean is also installed at the same time. After installation, you will have the main components in `<install_root>` as follows:

- **lib\JREngine.jar***
  Class library that contains the Catalog Bean.
- **lib\log4j-1.2.8.jar***
  Jar file used for the JReport Engine logging system.
- **bin\jslc.dat***
  License control file.
- **bin\report.ini***
  Configuration file.
- **help\samples\APICatalog\TestCatalogBean.java**
  The test tool of the Catalog Bean with a graphics user interface.

Catalog Bean classes

JReport Catalog Bean contains several classes: JRCatalog class, JRCatalogException class and ConnectionDesc class. Read this section to get detailed information.

JRCatalog class

The JReport Catalog is the principal class that manipulates and accesses a catalog. The application creates a JReport Catalog instance, and uses JReport Catalog methods to manipulate the catalog.

JReport Catalog has two properties:

- reportHome - The string that contains the installation path of JReport Catalog Bean.
- catName - The string that contains the full file path of the catalog. This path is used when loading or creating a catalog.

JRCatalogException class

JRCatalogException is the class that describes the error information. This exception is thrown when an error occurs. This exception describes many kinds of errors. The error code in the exception represents the exception type. The error code and error message can be viewed through public means.

ConnectionDesc class

ConnectionDesc is the class that describes the JDBC connection information. This class is used to create/modify connections and get connection information.

Reference: JReport Javadoc jet.bean.JRCatalog, jet.bean.JRCatalogException and jet.universe. ConnectionDesc classes in `<install_root>\help\api`, which includes all the documents generated by Javadoc.

Programming with the Catalog Bean

A catalog is used to contain all resources used for building reports in JReport Designer, and JReport Catalog Bean provides a programming API for manipulating and accessing JReport catalogs. JReport Catalog Bean allows to achieve functions such as, creating an empty catalog, adding objects to a catalog, removing objects from a catalog, and getting object description in a catalog.

Go through the following topics to further understand how to program with JReport Catalog Bean:

- Preparations
- Using the Catalog Bean
- An illustrative example
- Test tool program

Preparations

Before programming with the Catalog Bean, you need to import the Catalog Bean classes by adding the following import statements to the beginning of your program. The first two import statements are added to import the Catalog Bean, and exception classes. The third is added to import the connection description class.

```
import jet.bean.JRCatalog;
import jet.bean.JRCatalogException;
import jet.universe.ConnectionDesc;
```

Then you need to create an instance of the Catalog Bean. The constructor of the Catalog Bean has no parameter. After the object has been created, you should then set the report home to the object.

```
JRCatalog jrCatalog = new JRCatalog();
jrCatalog.setReportHome("c:\\jreport") ;
```

With the Catalog Bean object, you can manipulate and access many catalogs. You do not need to create a bean for each catalog.

You can load an existing catalog or create a new catalog. After a catalog has been created or loaded, you can manipulate it before closing it. The closeCatalog method closes a catalog, releases resource and clears up the temporary file. Also, the program requires using the closeCatalog method when operations on it have been completed.

Using the Catalog Bean

After making the required preparations, you can now use the Catalog Bean to perform tasks.

Creating a catalog

The method newCatalog creates an empty catalog according to the catalog name that is set in catName, and saves it to disk.

When creating a new catalog, the directory of the catalog name should exist and contain no catalog files. You should make sure of this before you start creating the new catalog.

```
Try
{
    jrCatalog.setCatName("c:\\a.cat");
    newCatalog();
}
catch(JRCatalogException e)
{
    System.out.println(e);
}
```

Loading an existing catalog

The method loadCatalog is used to load a catalog according to the catalog name that is set in catName.

```
try
{
    jrCatalog.setCatName("c:\\a.cat");
    loadCatalog();
}
catch(JRCatalogException e)
{
    System.out.println(e);
}
```

Manipulating a catalog

A catalog can be manipulated by the actions of adding, deleting and editing of objects within it. A catalog contains many kinds of objects. These objects include connections, tables, views, synonyms, file queries, stored procedures, queries, WHERE portions, formulas, summaries and parameters.

For example, you can modify the connection, the name, the URL, the user, the password and the driver.

```
try
{
    ConnectionDesc conDesc=new ConnectionDesc();
    conDesc.strName="Demo";
    conDesc.strURL="jdbc:oracle:thin:@myhost:1521:orcl";
    conDesc.strUser="system";
    conDesc.strPassword="manager";
    conDesc.strDriver="oracle.jdbc.driver.OracleDriver";
    jrCatalog.modifyConnection(conDesc);
}
catch(JRCatalogException e)
{
  System.out.println(e);
}
```

Getting information

You can get information about objects in a catalog via some methods.

For example, you can get the connection information via the method getConnectionDesc().

```
Try
{
    ConnectionDesc conDesc = bean.getConnectionDesc();
}
catch(JRCatalogException e)
{
    System.out.println(e);
}
```

Saving a catalog

Changes to a catalog are saved to a disk only when the method saveCatalog is used.

```
Try
{
    jrCatalog.saveCatalog();
}
catch(JRCatalogException e)
{
    System.out.println(e);
}
```

Handling exceptions

When an error occurs, the JRCatalogException is thrown out. Your program will catch the exception and handle the error.

The exception describes the error code and detailed message as follows:

- **Code: 0**
  Message: "System initialize failed."
  Description: The system initialization process failed. The Catalog Bean needs to be initialized to resolve the environment. This error is caused by reasons such as wrong report home, and bad installation.
  Detail: The reason why the initialization failed.
- **Code: 1**
  Message: "The catalog name is null."
  Description: The catalog name is null. The catalog name should be set before the loadReport() or newReport() methods are used.
  Detail: None.
- **Code: 2**
  Message: "Creating the catalog failed, because the specified directory does not exist or there is an existing catalog in that directory."
  Description: Cannot create a catalog in the specified directory, because the directory does not exist or there is a catalog in the directory.
  Detail: None
- **Code: 3**
  Message: "Cannot open the catalog file."
  Description: Cannot open the specified catalog file. The reason is file not found, file corrupt, format error or version mismatch.
  Detail: The reason why the opening of a catalog file failed.
- **Code: 4**
  Message: "Save catalog failed."
  Description: Cannot save the catalog file successfully. The cause is a file writing error, or a formula file save has failed.
  Detail: The reason why the saving of a catalog failed.
- **Code: 5**
  Message: "No catalog loaded."
  Description: The loadCatalog() or newCatalog() methods should be used before other methods, and closeCatalog() should be used in the end.
  Detail: None
- **Code: 6**
  Message: "The name of the WHERE portion is invalid or already exists."
  Description: The name specified is invalid or has already been used by another WHERE portion.
  Detail: None
- **Code: 7**
  Message: "The specified WHERE portion does not exist."
  Description: The specified WHERE portion does not exist.
  Detail: None.
- **Code: 8**
  Message: "A catalog can contain only one connection."
  Description: A catalog can contain only one connection, and this error occurs when adding a second connection.
  Detail: None.
- **Code: 9**
  Message: "The name of the connection is invalid or already exists."
  Description: The name specified is invalid or has already been used.
  Detail: None
- **Code: 10**
  Message: "Connection failure."
  Description: The JDBC connection failed when trying to create a connection to the database.
  Detail: The message of SQLException or ClassNotFoundException.
- **Code: 11**
  Message: "The name of the file query (customer SQL) is invalid or already exists."
  Description: The name of file query is either invalid or already exists.
  Detail: None.
- **Code: 12**
  Message: "Cannot create the file query."
  Description: Cannot create file query object.
  Detail: The reason why the file query cannot be created.
- **Code: 13**
  Message: "There is no connection in the catalog."
  Description: There is no connection in the catalog. If you are deleting or modifying a connection, the problem is that no such connection exists. If you are adding a file query, you cannot create it without a connection.
  Detail: None
- **Code: 14**
  Message: "The specified file query doesn't exist."
  Description: The specified file query does not exist.
  Detail: None.
- **Code: 15**
  Message: "Unknown Error."
  Description: The unexpected error.
  Detail: None
- **Code: 16**
  Message: "Invalid user ID or password"
  Description: The user ID or password is invalid.
  Detail: None
- **Code: 17**
  Message: "Only Tables or Views can be added into catalog."
  Description: The table type users specified is not supported.
  Detail: None
- **Code: 18**
  Message: "Close connection failure."
  Description: When closing connection, an error occurred.
  Detail: None

An illustrative example

In the following program, the Catalog Bean will be used to change the connection of a catalog:

```
try
{
// create catalog bean
JRCatalog jrCatalog = new JRCatalog();
// set report home
jrCatalog.setReportHome( "c:\\jrcbean");
// load catalog
jrCatalog.setCatName("c:\\a.cat">);
loadCatalog();
// modify connection
ConnectionDesc conDesc = new ConnectionDesc();
conDesc.strName = "Demo";
conDesc.strURL = "jdbc:oracle:thin:@myhost:1521:orcl";
conDesc.strUser = "System";
conDesc.strPassword = "Manager";
conDesc.strDriver = "oracle.jdbc.driver.OracleDriver";
jrCatalog.modifyConnection(conDesc);
// save catalog
jrCatalog.saveCatalog();
// close catalog
jrCatalog.closeCatalog();
}
catch(JRCatalogException e)
{
System.out.println(e);
}
```

Test tool program

TestCatalogBean.java located in `<install_root>\help\samples\APICatalog`, is a test tool for JReport Catalog Bean. It is primarily designed and built for you to test all the methods of JReport Catalog Bean.

The test tool program has a user interface, and all the functions of the Catalog Bean can be invoked using the menu items. Depending on success or failure of each test, you will get either a return result or an exception message respectively.

To run the JReport Catalog Bean test tool program, you should:

1. Compile the source code of TestCatalogBean.java. For example, JReport Designer is installed in `C:\JReport\Designer`, you can compile the it using the following command:

   ```
   C:\JReport\Designer\help\samples\APICatalog>javac -classpath "C:\JReport\Designer
   \lib\report.jar;C:\JReport\Designer\lib\sac.jar;C:\JReport\Designer\lib\JREngine.
   jar;" TestCatalogBean.java
   ```

2. Run the test tool:

   ```
   C:\JReport\Designer\help\samples\APICatalog>java -Dreporthome="C:\JReport
   \Designer" -classpath "C:\JReport\Designer\lib\JREngine.jar;C:\JReport\Designer\lib
   \sac.jar;C:\JReport\Designer\lib\report.jar;C:\JReport\Designer\lib\log4j-1.2.8.
   jar;C:\JReport\Designer\lib\hsqldb.jar" TestCatalogBean
   ```

   After activating the program, a window will be displayed.

3. Before starting other commands, you should first set the home path and the catalog name.

   Select **File** > **Set Home Directory** to set the home path and **File** > **Set Catalog Name** to set the catalog name.

4. Then, you can create a new catalog with the specified catalog name, or load an existing catalog in the same way.

5. The test tool also provides the source code of each method, and you can copy and paste it into your program from the text window.

JReport Exit functions

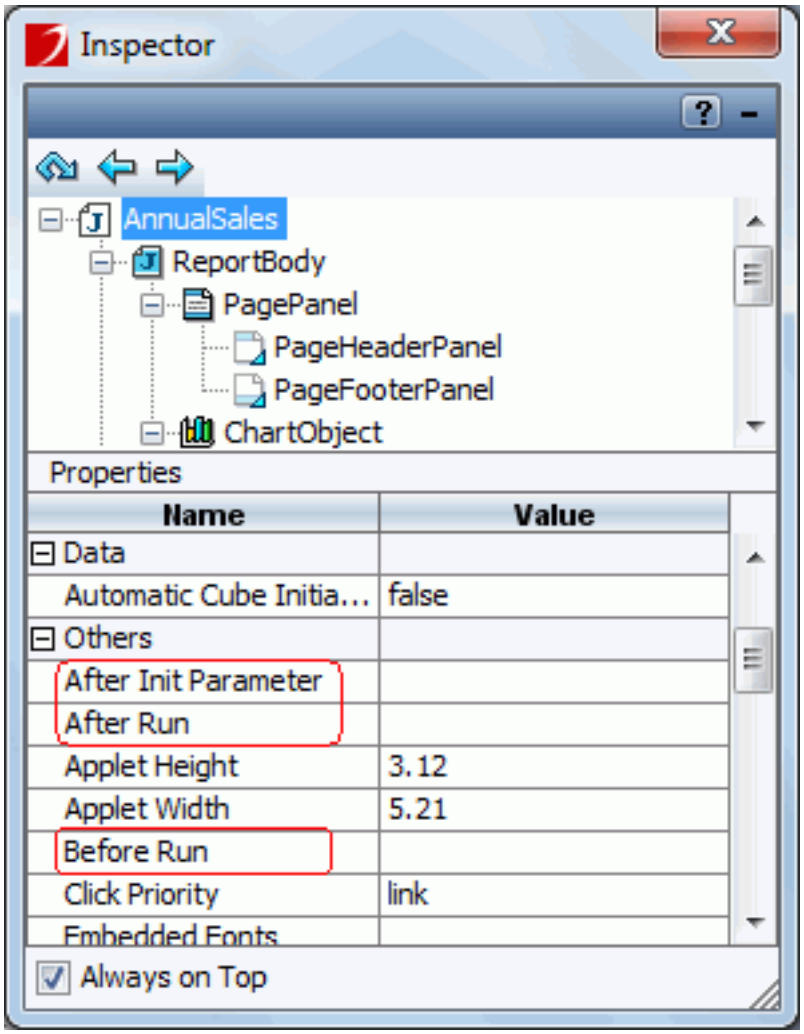
JReport has developed three Exit functions: After Init Parameter, After Run and Before Run. These functions enable you to develop an action to be called before, during and after the process of running a report. A status will then be returned. If the returned status is true, the JReport Engine will go on running. If false, the JReport Engine will stop at this point.

The following topics explain the JReport Exit functions in detail:

- Properties
- Executive interfaces
- Using the Exit functions

Properties

Open a report in JReport Designer. In the Report Inspector, select the node that represents the report tab and you will find the three Exit function properties: After Init Parameter, After Run and Before Run.



- **After Init Parameter**
  When running the report, after you enter parameters in the parameter dialog, this function allows you to call a program developed by yourself.

- **After Run**
  Allows you to call a program developed by yourself after running a report.

- **Before Run**
  Calls a program developed by yourself just before running a report.

Reference: For API information, see JReport Javadoc jet.util package and jet.exception package in `<install_root>\help\api`.

Executive interfaces

We provide three interfaces for you to implement:

jet.util.JURLExecutor

This interface provides the following method, which is used to get the status of the JReport Engine:

public boolean exec(String[] params) throws JURLExecuterException

- **Parameter**
  params - Array of strings, each element is defined and parsed by yourself. You can enter the PARAMETERs in the Report Inspector. If no PARAMETER is entered, params.length will be zero.
- **Return**
  Boolean value. If true, the JReport Engine will continue running. If false, JReport Engine will stop at this point.

jet.util.JURLEngineExecutor

This interface provides the following method, which is used to get the status of the JReport Engine:

public boolean exec(String[] params, EngineExecutor executor) throws JURLExecutorException

- **Parameter**
  params - Array of strings, each element of which is defined by yourself and also parsed by yourself. You can enter the PARAMETERs in the Report Inspector. If no PARAMETERs are entered, then params. length is zero.
  executor - Executor of EngineExecutor, which is another interface JReport provides for exporting a report to HTML, TXT or PDF format within the Exit functions.
- **Return**
  boolean value. If true, the JReport Engine will continue running. If false, the JReport Engine will stop at this point.

jet.util.EngineExecutor

This interface provides three methods:

- **public boolean exportToHtml(String htmlFileName, boolean bChartApplet, boolean isMultiFile, boolean bUsingTable, boolean bHyperlink, boolean bPageNum, boolean bAbsolute, int iBrowser)**

  This method is used to export the report result to HTML format.

  - **Parameter**
    HtmlFileName - Name of the HTML format result file with full path.
    bChartApplet - A Boolean value which specifies whether or not a chart in an HTML file is an applet.
    isMultiFile - A Boolean value which specifies whether or not the HTML is generated to one or multiple files.
    bUsingTable - A Boolean value which specifies whether or not to use the HTML table format for exporting HTML files.
    bHyperLink - A Boolean value which specifies whether or not to generate hyperlinks.
    bPageNum - A Boolean value which specifies whether or not to generate page number.
    bAbsolute - A Boolean value which specifies whether or not the font size is absolute or relative.
    iBrowser - An int value which specifies whether or not the web browser is IE.
  - **Return**
    Boolean value for exporting status. If true, the report has been successfully exported and vice versa.

- **public boolean exportToPdf(String pdfFileName)**

  This method is used to export the report result to PDF format.

  - **Parameter**
    PdfFileName - The name of the PDF format result file with a full path.
  - **Return**
    Boolean value for exporting status. If true, the report has been successfully exported and vice versa.

- **public boolean exportToText(String textFileName, boolean isNormalText, boolean isRepeat, char delimiter)**

  This method is used for exporting the report result to text format.

  - **Parameter**
    TextFileName - Name of the TEXT format result file with full path, where "true/false" sets whether or not the text file is of normal text format or of standard data format. A standard data format text file is a text file where each row represents a single record in a report. It can be used as a text data source for exchanging data with other applications.
    isNormalText - A Boolean value which specifies whether or not a text file is in normal text format.
    isRepeat - A Boolean value which specifies whether or not the contents will be repeated.
    delimiter - A character which is be used in SDF (Standard Data Format). The delimiter will only be used when the "true/false" parameter is set to false. Delimiters can be ',' (the CSV format) or any other character. The default is ' ' (a blank), which is the SSV format.
  - **Return**
    Boolean value for exporting status. If true, the report has been successfully exported and vice versa.

Reference: For API information, see JURExecutor, JUREngineExecutor and EngineExecutor interfaces in the JReport Javadoc jet.util package located at `<install_root>\help\api`.

Using the Exit functions

This section provides you with two examples to further understand the Exit functions.

Example 1: Using parameters in the Exit functions

In this example, we provide a simple application which will run at all the three times, Before Run, After Init Parameter, and After Run.

1. Develop your Java files to implement the methods. In the example, copy the following code and save it as **testa.java** to `C:\JReport\Designer\help` (here it is assumed that you have installed your JReport Designer to the default directory).

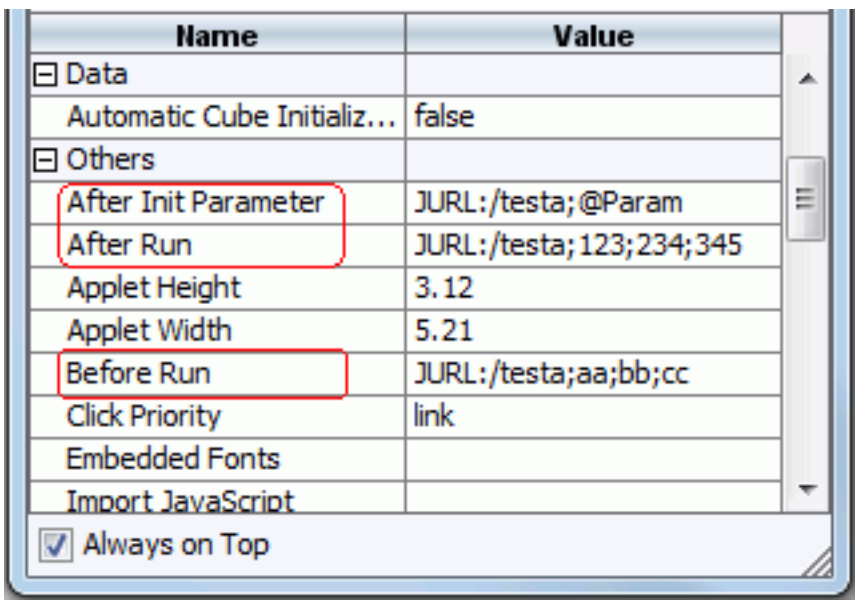
```
import java.io.*;
import java.awt.*;
import java.net.*;
import jet.util.*;

public class testa implements JURLExecuter
{
        public boolean exec(String[] params)
        {
                System.out.println("testa : ");
                for (int i = 0; i < params.length; i++)
                {
                        System.out.print("\"" + params[i] + "\",");
                }
                System.out.println();
                return true;
        }
}
```

2. Compile testa.java to generate the class file testa.class in `C:\JReport\Designer\help`.
3. Modify the batch file setenv.bat in `C:\JReport\Designer\bin` by appending the path `C:\JReport\Designer\help` into the batch file's ADDCLASSPATH variable:

   `set ADDCLASSPATH=%JAVAHOME%\lib\tools.jar;`**`C:\JReport\Designer\help;`**

4. Start JReport Designer, then open a report with parameter.
5. In the Report Inspector, select the node that represents the report tab and specify the class and give parameters for the three functions, for example:



Name	Value
⊟ Data	
Automatic Cube Initializ...	false
⊟ Others	
After Init Parameter	JURL:/testa;@Param
After Run	JURL:/testa;123;234;345
Applet Height	3.12
Applet Width	5.21
Before Run	JURL:/testa;aa;bb;cc
Click Priority	link
Embedded Fonts	
Import JavaScript	
☑ Always on Top	

JURL:/ - The standard format, and followed by the class name.
testa - The class name in this example.
aa;bb;cc and **123;234;345** - The string type parameters of the class, which are divided by semicolons.
Param - A predefined JReport parameter used in the current report, which is introduced by the symbol @.

As shown above, two types of Exit function parameters are supported, they are:

- String type parameters, such as aa, C:\ and so on. In this case, the string will be passed directly to the Exit function.
- JReport predefined parameters used in the current report, which are introduced by the @ symbol. In this case, the parameter value will be passed to the Exit function.

6. Run the report, the report engine will call the class that you specified in the Report Inspector.

Example 2: Using formulas in the Exit functions

JReport allows you to use formulas in the Exit functions. To do this, you can use the Data Container Link function which can return value of any formula to a parameter. And using the parameter in the Exit functions, you will get the value of the formula in the Exit functions.

1. Create a formula and insert it into a report. It should be put in the child data container.
2. Create a parameter with the same data type of the formula, leaving other options blank, and then insert it into the parent data container.
3. Right-click the parent data container and click **Data Container Link** on the shortcut menu.
4. In the Data Container Link dialog, click the **Return Value** tab, add the formula in the Field in Child Data Container to the Return Value box, then click **OK**.

   Now, you have finished passing value of the formula to a parameter.
5. Use this parameter in the Exit functions with the method described in Example 1, and then the final value of formula will be returned in the Exit functions.

Notes:

- When a JReport parameter is used as the parameter of the Before Run function, no JReport parameter value will be returned, and the returned value will be the JReport parameter name with the @ symbol.
- For the JReport parameter that are used as Exit function parameters, they must be used in the report. Otherwise, no value will be returned.
- If you want to publish the report to JReport Server and run it there, you should publish both the report and the catalog file to JReport Server. However, do not forget to add the path to the `-classpath` option in the batch file that starts JReport Server.
- Using formulas in Exit functions can also be realized by using returning values in subreport. Thus, you need to pass value of a formula in the subreport to a parameter in the primary report.

Accessibility

JReport supports accessibility attributes and a built-in accessible JReport Server console. The implementation standard is based on HTML specification 4.01 http://www.w3.org/TR/WCAG10-HTML-TECHS/ and information on Section 508 Standards: http://www.section508.gov and http://www.access-board.gov.

Making HTML format report result accessible

When designing a report in JReport Designer, you can add the accessibility related HTML attributes to the report elements in order to make the HTML format result more readable and accessible. Those attributes are provided in the Accessibility category of the Report Inspector.

To add accessibility to a report in the HTML format result:

1. Predefine necessary accessibility attributes when designing the report in JReport Designer.

   When a report element is selected, the corresponding accessibility attributes are available in the Report Inspector.

2. Enable Section 508 compliant output when exporting the report to HTML format.

   In the HTML export option UI, check the option **Section 508 Compliant Output**.

   If you only want to convert table/crosstab components into HTML data table in the HTML format report result, check the option **Use HTML Data Table**.

   The above two options are available on all HTML export option UIs in JReport Designer, JReport Server and Page Report Studio.

The accessible version of JReport Server

JReport Server enables the disabled users to visit the accessible version with simplified functionality to read reports by reader agent or other assistive tools. In the accessible version, reports are displayed in HTML format with accessibility attributes, and table/crosstab components are output as HTML data table.

JReport Server provides an option - Use Accessible Version, for setting the accessible version of JReport Server as the default portal UI for an end user after the user logs in. The default port for accessible version is 8888 which is also the default port for JReport Console page in normal version. That is to say, the Use Accessible Version option controls switching between normal version and accessible version of JReport Server UI when logging onto port 8888. By default this option is unselected and end users are directed to the JReport Console page in normal version.

Activating the accessible version

The administrator can activate the accessible version for all users or for an individual user:

- **To enable for all users:**
  1. Log onto the JReport Administration page.
  2. Click **Profile** on the system toolbar and then select **Customize Server Preferences** from the drop-down menu.
  3. Select the **Advanced** tab.
  4. Check the **Use Accessible Version** option, then click **OK** to apply the change.
- **To enable for an individual user:**
  1. Log onto the JReport Administration page.
  2. Click **Security** on the system toolbar and then select **User** from the drop-down menu.
  3. Locate the wanted user ID, then click the **Preference** link for the user.
  4. In the Preference dialog, go to the **Customize Server Preferences** > **Advanced** tab.
  5. Check the **Use Accessible Version** option, then click **OK** to save the change.

And a non-administrator user can also enable the accessible version by self on the JReport Console page in the following way:

1. Log onto the JReport Console page and click **Profile** on the system toolbar.
2. Click **Customize Server Preferences** on the task bar of the Profile page, then select the **Advanced** tab.
3. Check the **Use Accessible Version** option, then click **OK** to save the change.

After a user has been enabled the access to Accessible Version, the user will be directed to the accessible version when he/she logs in next time. With the help of reader agent the user can navigate through the server resource to view the target report.

Main options in the accessible version UI

Directory Path

Displays the current directory where you have come to follow the server resource tree.

Up to Higher Level Directory

Goes to the parent level folder.

Up to Top Level Directory

Goes to the default portal page.

Leave Accessible Version

Redirects to the normal version of JReport Console page with full functionalities. Once you leave the accessible version, there is no way to return unless you re-log in. However, it is not recommended for the disabled to leave the accessible version unless having others' help.

Select User Directory

You can choose to open either the My Reports folder or the Public Reports folder.

Catalog

Lists the catalogs used in the current directory.

Report List

Lists the folders and report and report tabs contained in the current directory. You can click the hyperlinks in the Name column to open them.

Dialog Reference

This chapter covers the dialogs that you will come across when using JReport Server, providing descriptions and explanations of the options available. By referencing this chapter, you will find JReport Server much easier and more convenient to use.

In this chapter, the following types of dialogs are covered:

- JReport Server dialogs
- Page Report Studio dialogs
- Web Report Studio dialogs
- JDashboard dialogs
- Visual Analysis dialogs

JReport Server dialogs

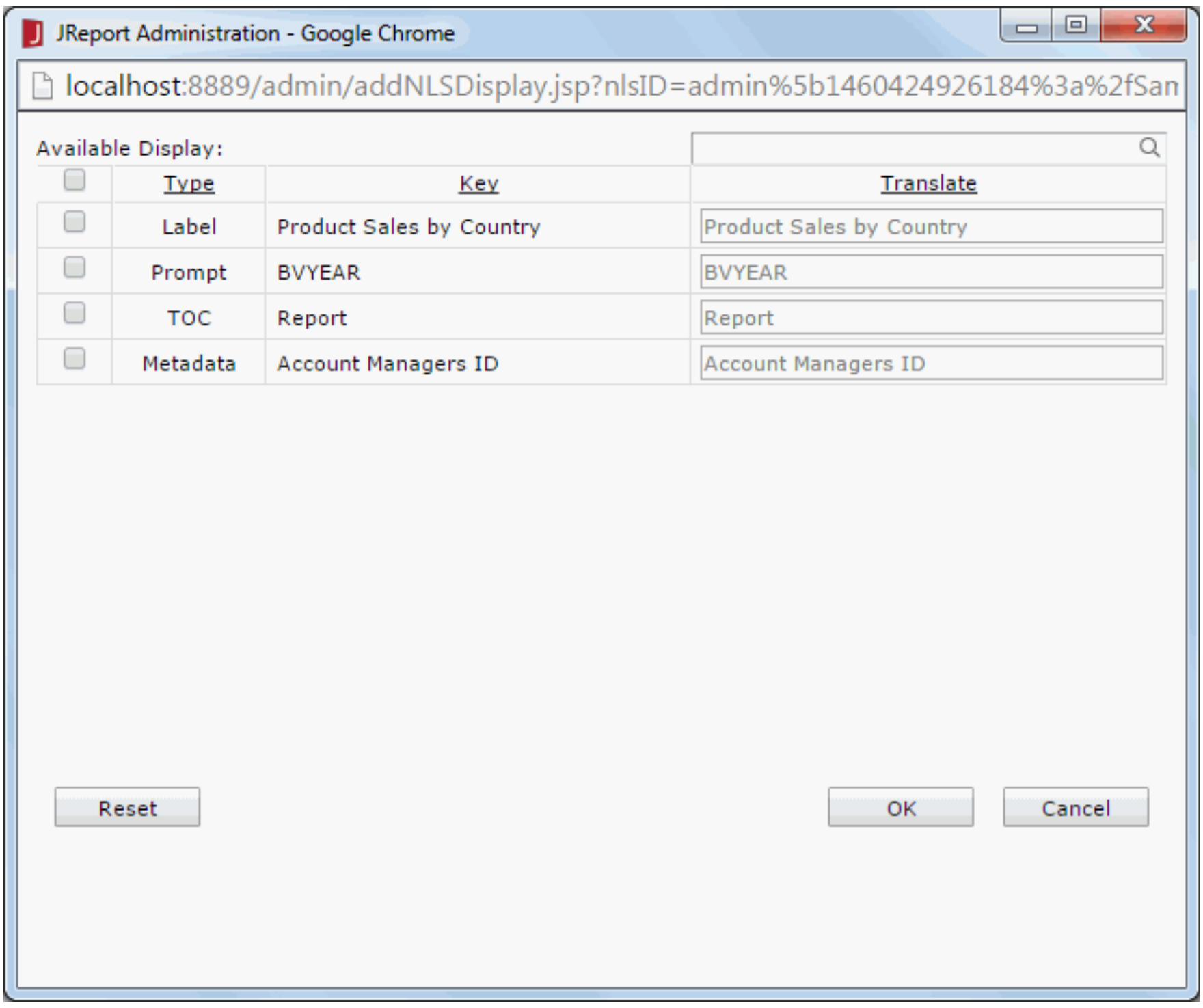
There are two classes of dialog in JReport Server. They are distinguished by the types of users who can gain access to them. The JReport Console page dialogs are available to general users, while the JReport Administration page dialogs are only accessible to users who have been assigned the administrator's role.

The following are dialogs of the JReport Server, which are listed in alphabetic order:

- Add Display dialog
- Add Dynamic Connection dialog
- Add Dynamic Display Names dialog
- Add Font dialog
- Add Format dialog
- Add Language dialog
- Add to Global NLS dialog
- Advanced Run dialog
- Auditing dialog
- Cache Configuration dialog
- Cache dialog
- Calendar dialog
- Catalog Properties dialog
- Cluster dialog
- Configuration dialog
- Create Organization dialog
- Cube Configuration dialog
- Cube dialog
- Customize Waiting Page dialog
- Dashboard Properties dialog
- Data dialog
- Edit Dynamic Connection dialog
- Edit Dynamic Display Names dialog
- Encrypt dialog
- Enter Parameter Values dialog
- Enter Values dialog
- Folder Properties dialog
- Global NLS dialog
- JDashboard Profile dialog
- Library Component Properties dialog
- New Cache dialog for scheduling
- New Cache dialog for selecting queries
- New Cube dialog for scheduling
- New Cube dialog for selecting business view
- New Custom Field dialog
- New Dashboard Listener dialog
- New Dynamic Security dialog
- New Group dialog
- New Realm dialog
- New Role dialog
- New Schedule dialog
- New User dialog
- NLS Editor dialog
- Page Report Studio Profile dialog
- Parameter Settings dialog
- Profile dialog
- Publish to Local Server dialog
- Publish to Remote Server dialog
- Report Properties dialog
- Resource Allocation dialog
- Resource Tree NLS dialog
- Result Properties dialog
- Schedule dialog
- Security dialog
- Select Another Catalog dialog
- Select Business View Elements dialog
- Select Catalog dialog
- Select Dynamic Display Name File dialog
- Select Folder dialog
- Select Language Source dialog
- Select Queries dialog
- Select Target dialog
- Set Alias dialog
- Set Permissions dialog
- Sign dialog
- Triggers dialog
- Upload Dynamic Security dialog
- Visual Analysis Template Properties dialog

Add Display dialog

This dialog appears when you click the button + in the Display tab of the NLS Editor dialog. It helps you to add available display text keys in the specified catalog/report/library component for editing NLS.



Available Display

Lists all the available display text keys in the specified catalog/report/library component.

Quick search toolbar

Searches for the keys that contain the specified text. After typing the text, click the search button on the toolbar or press Enter on the keyboard to start searching.

Checkbox

Specifies whether to add the display text keys to edit NLS. Select the checkbox on the column header to select all keys.

Type

Lists types of display text for different objects. You can click the underlined column header to sort the items by type in an ascending or descending order.

- **Column**
  This type is only for reports running in Page Report Studio. It is the type of display text of columns.
- **DisplayName**
  Type of display text of object display name.
- **Label**
  Type of display text of label, some web controls and UDOs.
- **Metadata**
  Type of display text of metadata. Metadata mainly refers to catalog resources, such as table/view columns, business views, formulas, summaries, parameters, and so on.
- **Prompt**
  Type of display text of parameter prompt value.
- **Title**
  Type of display text of filter control and library component.
- **TOC**
  Type of display text on the TOC tree.

Key

Lists keys to indicate the objects in the original language. You can click the underlined column header to sort the items by key in an ascending or descending order.

Translate

Specifies the translation of the keys in the target language. You can click the underlined column header to sort the items by translation in an ascending or descending order.

Reset

Resets the dialog to its initial status.

OK

Adds the selected keys to the Display tab of the NLS Editor dialog.

Cancel

Cancels the action and exits the dialog.

Help

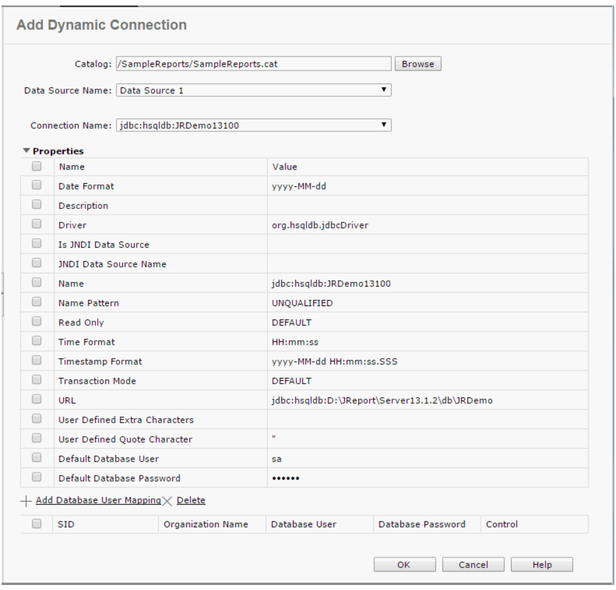
Displays the help document about this feature.

Related topics:

- Editing local NLS

Add Dynamic Connection dialog

The dialog appears when you click the Add Dynamic Connection link in the Dynamic Connections panel. It helps you to create a dynamic connection.



Catalog

Specifies the catalog in which you would like to create a dynamic connection. Type the catalog with the full resource path in the text box, for example, `/SampleReports/SampleReports.cat`, or click the Browse button to select a catalog in the Select Catalog dialog.

Data Source Name

Specifies the data source in which you would like to create a dynamic connection in the catalog from the drop-down list.

Connection Name

Specifies a connection to connect to the data source from the drop-down list.

Properties

Displays the properties of the database connection. You can click it to expand or collapse the property table.

Here you can change some properties of the database connection. The table shows the values of the original connection properties initially.

- **Checkbox**
  After you select the checkbox for a property, you will be able to change the property value. Select the checkbox on the column header and the checkboxes for all the properties will be selected. Initially all the properties are unchecked and uneditable. Whenever a property is unchecked, it always shows the original value regardless of whether you have changed the value or not.
- **Name**
  Displays the property names.
- **Value**
  Specifies the property values.

There are the following properties:

- **Date Format**
  Specifies the default Date format corresponding to the database.

  Date Format and Timestamp Format

Symbol	Meaning	Presentation
y	year	Number
M	month	Number
d	day in month	Number
H	hour in day (0~23)	Number
h	hour in am/pm (1~12)	Number
m	minute in hour	Number
s	second in minute	Number
S	millisecond	Number

  Example (using the US Locale):

  "yyyyy.MMMMM.dd hh:mm aaa" ->> 1996.July.10 12:08 PM

  Note: The Date and Timestamp format JReport supports follows that of Java. Refer to the Java API Specification java.text package DateFormat interface.
- **Description**
  Specifies the description of the connection.
- **Driver**
  Specifies the class name of the JDBC driver such as oracle.jdbc.driver.OracleDriver.
- **Is JNDI Data Source**
  Specifies whether it is a JNDI data source.
- **JNDI Data Source Name**
  Specifies the JNDI data source name when the Is JNDI Data Source property is true.
- **Name**
  Specifies the name of the connection.
- **Name Pattern**
  Specifies whether or not catalog or schema is used in data manipulation. Choose an option from the drop-down list.
  - **Unqualified** - Neither catalog nor schema is included in data manipulation. Example: SELECT t.c FROM t
  - **Qualified_2part** - Uses schema in data manipulation. Example: SELECT schema.t.c FROM schema.t
  - **Qualified_3part**- Uses both catalog and schema in data manipulation. Example: SELECT catalog.schema.t.c from catalog.schema.t
- **Read Only**
  Specifies the mode to open the connection to the JDBC data source. The initial setting is Default which uses the mode specified by the DBMS Administrator which could be read only or read & write. Choose an option from the drop-down list.
  - **Default** - May be read & write or read only depending on the DBMS default setting.
  - **Read only** - Allows the driver to optimize performance for reporting which does not need to write to the DBMS.
  - **Read & write** - Opens the DBMS with updates enabled which requires more processing to ensure concurrency control.
- **Time Format**
  Specifies the default time format corresponding to the database.
- **Timestamp Format**
  Specifies the default timestamp format corresponding to the database.
- **Transaction Mode**
  Specifies the transaction mode for the connection. Choose an option from the drop-down list.
  - **Default** - Indicates the transaction information cannot be got from JDBC connection.
  - **None** - Indicates that transactions are not supported.
  - **Read_uncommitted** - Dirty reads, non-repeatable reads and phantom reads can occur. This mode will speed up the transaction of the catalog.
  - **Read_committed** - Dirty reads are prevented; non-repeatable reads and phantom reads can occur.
  - **Repeatable_read** - Dirty reads and non-repeatable reads are prevented; phantom reads can occur.
  - **Serializable** - Dirty reads, non-repeatable reads and phantom reads are prevented.
- **URL**
  Specifies the JDBC URL which establishes the connection to the database, for example, jdbc:oracle:thin:@localhost:1521:ORCL.
- **User Defined Extra Characters**
  Specifies extra characters of user defined quote qualifier.
- **User Defined Quote Characters**
  Specifies the quote character of user defined quote qualifier.
- **Default Database User**
  Specifies the user name for connecting to the database, which is determined by the database DBA. Null means using the default database user name.
- **Default Database Password**
  Specifies the password for connecting with the database, which is determined by the database DBA. Null means using the default database password.

Add Database User Mapping

Adds a new database user mapping. Click to create a new line at the end of the table.

Delete

Deletes the selected database user mappings.

Database user mapping table

Checkboxes are used to specify whether or not to select the database user mapping. Select the checkbox on the column header to select all database user mappings. After you select the database user mappings, you can then delete them if you do not want them.

- **SID**
  Specifies the security identifier (SID). A SID can be a group, role or user in the JReport Server security system. You can select a value from the drop-down list or input the value in the text box. The **x** in the text box is used to clear the input text. You can define at most one database user mapping for an SID within a dynamic connection.
- **Organization Name**
  Double-click in the text box and then select an organization from the drop-down list. You can first specify the organization and then the SID. The column is available to system admin when the organization feature is enabled.
- **Database User**
  Specifies the database user name. Null means using the default database user name.
- **Database Password**
  Specifies the database password. Passwords are masked.
- **Control**
  - **Test Connection**
    Tests whether the connection configuration works by using the database use rname and password.

OK

Creates the dynamic connection and exits the dialog.

Cancel

Cancels the creation of a dynamic connection and closes the dialog.

Help

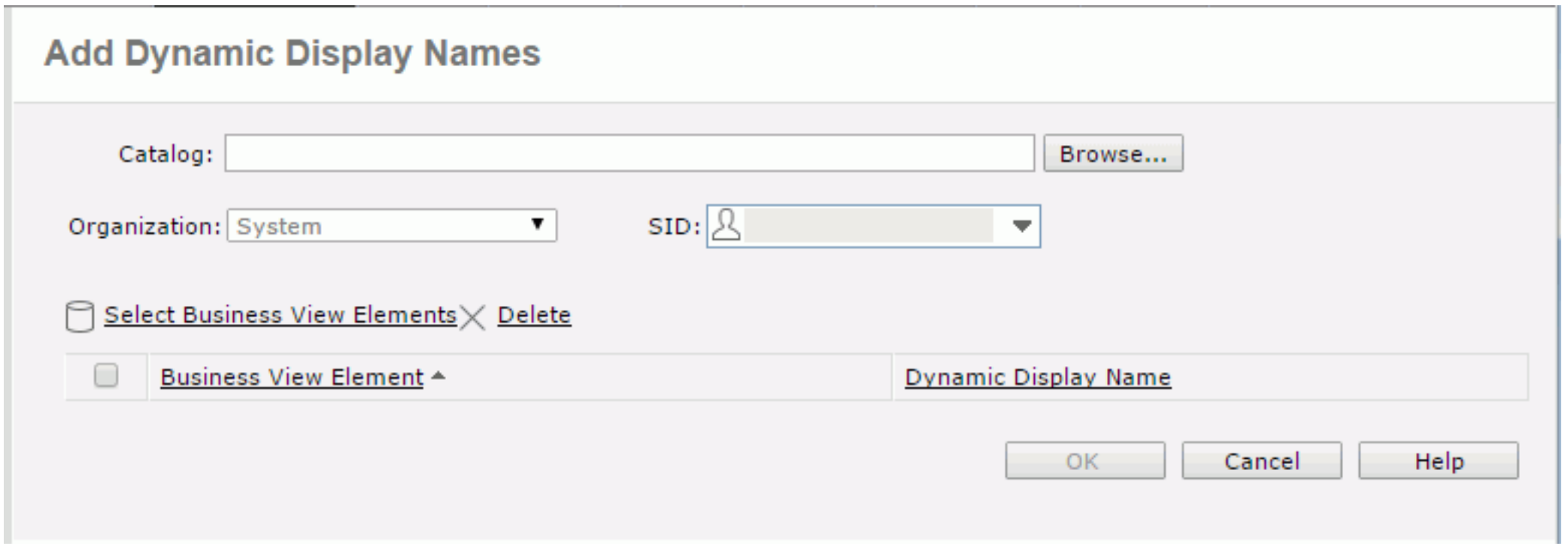
Displays the help document about this feature.

Related topics:

- Configuring dynamic connections

Add Dynamic Display Names dialog

This dialog appears when you click the Add Dynamic Display Names link in the Dynamic Display Names panel. It helps you to add dynamic display names for business view elements in a catalog for a specified SID (security identifier).



Catalog

Specifies the catalog that contains the required business views for which you want to define dynamic display names.

Type the catalog with the full resource path in the text box, for example, `/SampleReports/SampleReports.cat`, or click the **Browse** button to select a catalog in the Select Catalog dialog.

Organization

The option is available when the organization feature is enabled. System admin can select an organization from the drop-down list. For an organization admin, the organization of the admin is selected by default and cannot be changed.

SID

Specifies the SID for which you want to define dynamic display names. An SID can be a group, role, or user in the JReport Server security system. Select a value from the drop-down list. The **x** button in the text box is used to clear the current value. Empty means all the users (when the organization feature is enabled, empty means all the users in the specified organization).

Select Business View Elements

Opens the Select Business View Elements dialog to select business view elements in the specified catalog for which to define dynamic display names.

Delete

Deletes the selected business view elements.

Business view elements table

Checkboxes are used to specify whether or not to select the business view elements. Select the checkbox on the column header to select all the business view elements. After you select the business view elements, you can then delete them if you do not want them. You can also click the underlined column header name to sort the elements either by their qualified names or display names.

- **Business View Element**
  Displays the qualified names of the business view elements with the path in the catalog.
- **Dynamic Display Name**
  Shows the display names of the business view elements. The display names are editable. Double-click in a name text box to enter the editing mode and then change the name. The **x** button in the text box is used to clear the text.

OK

Applies the dynamic display names and exits the dialog.

Cancel

Cancels adding the dynamic display names and closes the dialog.

Help

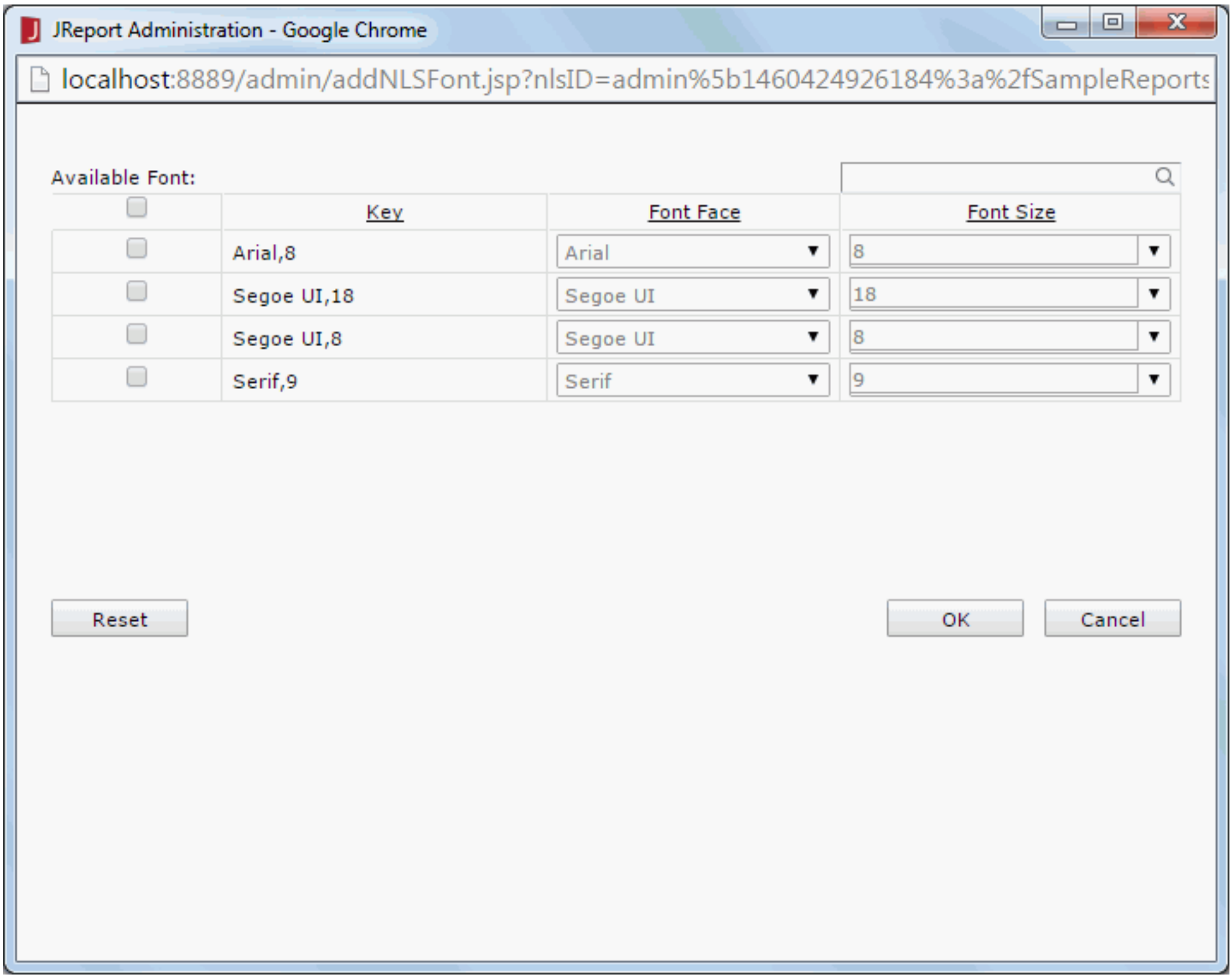
Displays the help document about this feature.

Related topics:

- Managing dynamic display names of business view elements

Add Font dialog

This dialog appears when you click the button + in the Font tab of the NLS Editor dialog. It helps you to add available font keys in the specified catalog/report/library component for editing NLS.



Available Font

Lists all available font keys in the specified catalog/report/library component.

Quick search toolbar

Searches for the keys that contain the specified text. After typing the text, click the search button on the toolbar or press Enter on the keyboard to start searching.

Checkbox

Specifies whether to add the font keys to edit NLS. Select the checkbox on the column header to select all keys.

Key

Lists keys to indicate the fonts in the original language. You can click the underlined column header to sort the items by key in an ascending or descending order.

Font Face

Specifies the font faces for the keys in the target language. You can click the underlined column header to sort the items by font face in an ascending or descending order.

Font Size

Specifies the font sizes for the keys in the target language. You can click the underlined column header to sort the items by font size in an ascending or descending order.

Reset

Resets the dialog to its initial status.

OK

Adds the selected keys to the Display tab of the NLS Editor dialog.

Cancel

Cancels the action and exits the dialog.

Help

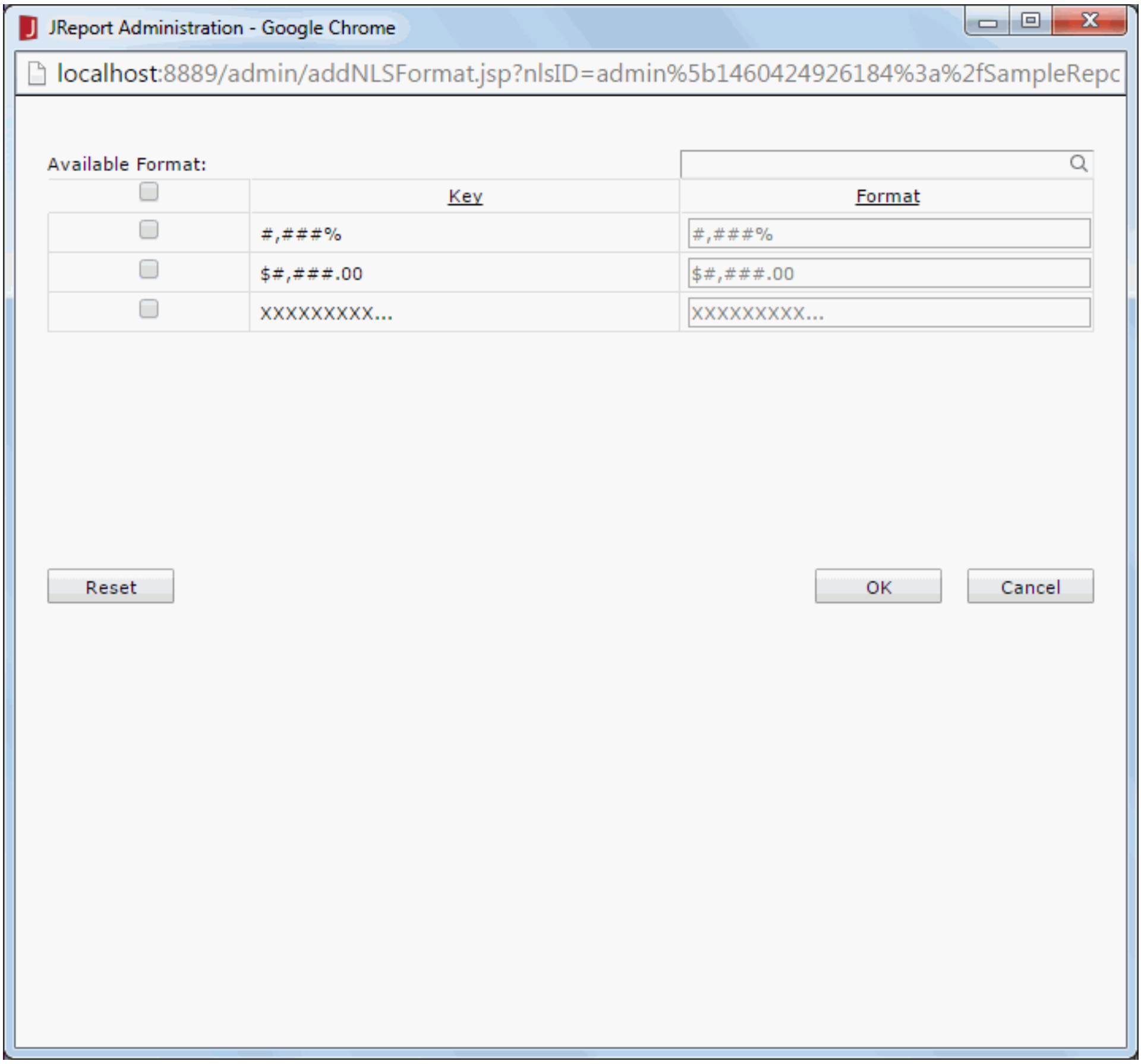
Displays the help document about this feature.

Related topics:

- Editing local NLS

Add Format dialog

This dialog appears when you click the button + in the Format tab of the NLS Editor dialog. It helps you to add available format keys in the specified catalog/report/library component for editing NLS.



Available Format

Lists all available format keys in the specified catalog/report/library component.

Quick search toolbar

Searches for the keys that contain the specified text. After typing the text, click the search button on the toolbar or press Enter on the keyboard to start searching.

Checkbox

Specifies whether to add the format keys to edit NLS. Select the checkbox on the column header to select all keys.

Key

Lists keys to indicate the objects in the original language. You can click the underlined column header to sort the items by key in an ascending or descending order.

Format

Specifies the formats for the keys in the target language. You can click the underlined column header to sort the items by format in an ascending or descending order.

Reset

Resets the dialog to its initial status.

OK

Adds the selected keys to the Display tab of the NLS Editor dialog.

Cancel

Cancels the action and exits the dialog.

Help

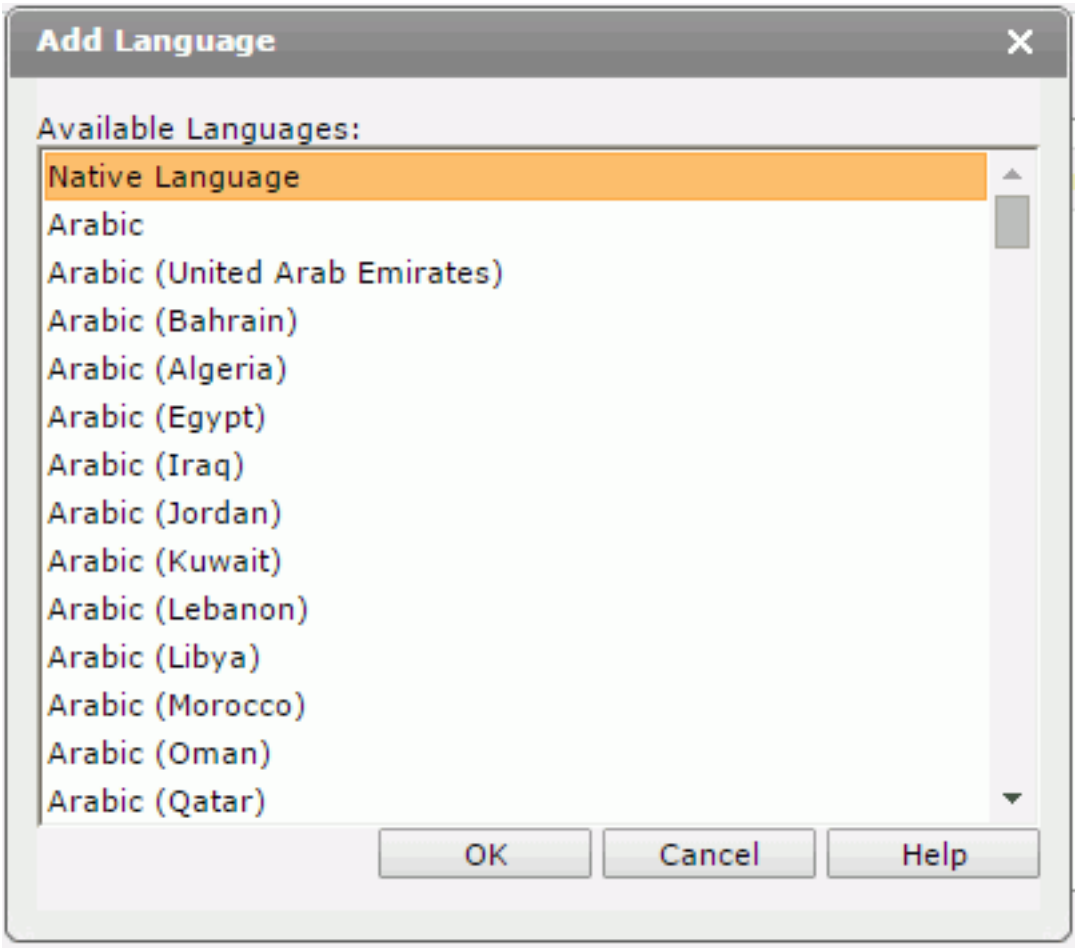
Displays the help document about this feature.

Related topics:

- Editing local NLS

Add Language dialog

This dialog appears when you click the Add button + in the NLS Editor dialog when editing NLS for a catalog, report or library component, or click OK in the Select Language Source dialog with Languages Supported by JReport selected. It helps you to add one or more languages, in which reports can be run.



Available Languages

Lists all the languages that are available for you to choose. Select the required languages from the box.

OK

Adds the selected languages and closes the dialog.

Cancel

Cancels the action and exits the dialog.

Help

Displays the help document about this feature.

Add to Global NLS dialog

This dialog is displayed when you click the Add to Global NLS button in the NLS Editor dialog to add the selected display, format or font items to the target language's global NLS resource library, but some of the selected items already exist there. It varies according to the type of the selected items.

The dialog is not available to organization admin.

OK

Adds the selected items to the target language's global NLS resource library and exits the dialog.

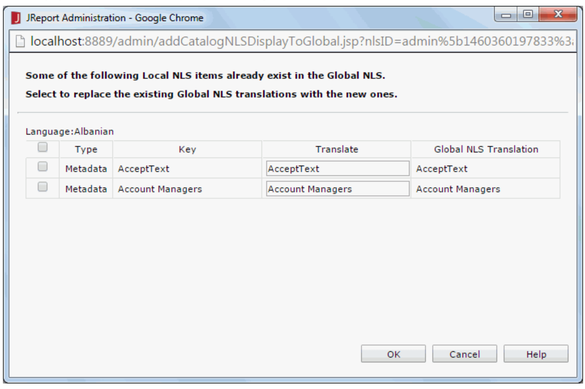
Cancel

Cancels the action and exits the dialog.

Help

Displays the help document about this feature.

When display items are to be added, options in the dialog are as follows.



Language

Displays the target language into which the display text is translated.

Checkbox

Specifies the items you want to add to the target language's global NLS resource library. Check the checkbox on the column header to select all the items.

Type

Lists types of display text for different objects.

- **Label**
  Type of display text of label and some web controls.
- **Column**
  This type is only for page reports running in Page Report Studio. It is the type of display text of columns.
- **Prompt**
  Type of display text of parameter prompt value.
- **TOC**
  Type of display text on the TOC tree.

Key

Lists keys to indicate the objects in the original language.

Translate

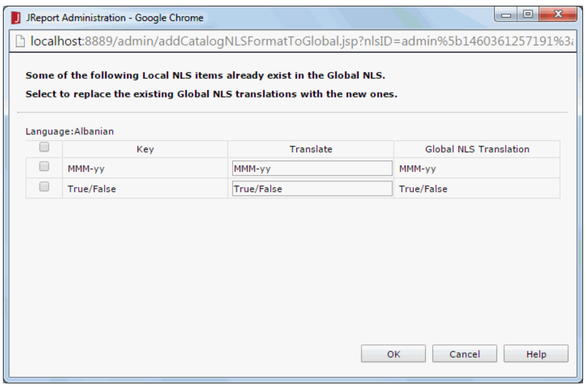
Specifies the translation for the keys in the target language.

Global NLS Translation

Displays the text to which the keys are translated in the target language's global NLS resource library. There are two circumstances:

- If the translation is Null for any key, it means the key doesn't exist in the target language's global NLS resource library yet. Check the checkbox ahead of the key if you want to add it.
- If any key already exists in the target language's global NLS resource library, and you have provided a new translation for it in the Translate column, when you choose to add this display item, the existing global NLS translation for the corresponding key will be replaced by the new one.

When format items are to be added, options in the dialog are as follows.



Language

Displays the target language into which the format properties will be added.

Checkbox

Specifies the items you want to add to the target language's global NLS resource library. Check the checkbox on the column header to select all the items.

Key

Lists keys to indicate the formats in the original language.

Translate

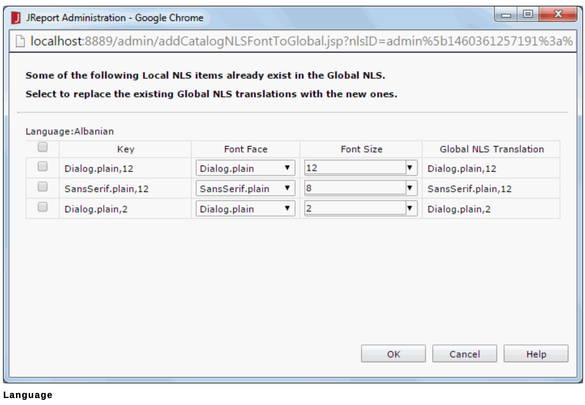
Specifies the formats for the keys in the target language.

Global NLS Translation

Displays the format properties for the keys in the target language's global NLS resource library. There are two circumstances:

- If the translation is Null for any key, it means the key doesn't exist in the target language's global NLS resource library yet. Check the checkbox ahead of the key if you want to add it.
- If any key already exists in the target language's global NLS resource library, and you have specified a new format for it in the Format column, when you choose to add this format item, the existing global NLS translation for the corresponding key will be replaced by the new one.

When font items are to be added, options in the dialog are as follows.



Language

Displays the target language to which the font properties will be applied.

Checkbox

Specifies the items you want to add to the target language's global NLS resource library. Check the checkbox on the column header to select all the items.

Key

Lists keys to indicate the fonts in the original language.

Font Face

Specifies the font faces for the keys in the target language.

Font Size

Specifies the font sizes for the keys in the target language.

Global NLS Translation

Displays the font properties for the keys in the target language's global NLS resource library. There are two circumstances:

- If the translation is Null for any key, it means the key doesn't exist in the target language's global NLS resource library yet. Check the checkbox ahead of the key if you want to add it.
- If any key already exists in the target language's global NLS resource library, and you have specified a new font face and font size for it in the Font Face and Font Size columns, when you choose to add this font item, the existing global NLS translation for the corresponding key will be replaced by the new one.

Advanced Run dialog

To access the Advanced Run dialog, on the JReport Console > Resources page, browse to the report you want to run in Advanced mode, then do either of the following:

- Select the row the report is in, then on the task bar of the Resources page, click Run > Advanced Run.
- Select the report row, right-click in the row and select Advanced Run from the shortcut menu.
- Put the mouse pointer over the report row and click the Advanced Run button on the floating toolbar.

There are five tabs in this dialog, which are listed below. You can set the settings to your requirements in each tab.

- General tab
- Parameter tab
- Format tab
- Archive tab
- Duration tab

Back

Goes back to the left tab.

Next

Goes to the right tab.

Finish

Applies the settings.

Cancel

Closes the dialog and discards any changes.

Help

Displays the help document about this feature.

Related topics:

- Running reports - advanced running

General tab

This tab allows you to specify the general information about a report.

Select Report Tabs

The section is available for page reports only. It allows you to specify the report tabs which you want to run. The selected report tabs will be run in the list order. If the page report has only one report tab, it is selected by default.

Moves the specified report tab one step up.

Moves the specified report tab one step down.

Select Dynamic Connection

The section is available when there are multiple dynamic connections for the current log-in user. It allows you to specify a dynamic connection.

Data Source

Displays the data source name in the catalog.

Connection Name

Displays the dynamic connection name.

Connection

Specifies a dynamic connection from the drop-down list.

Connection Properties

Shows the information of the selected dynamic connection, which is read only. For details about the propeties, refer to Edit Dynamic Connection dialog.

Report Information

Specifies other report information.

Report

Lists the report information.

Catalog

Lists the catalog information.

Select Another Catalog

Specifies another catalog for the report in the Select Another Catalog dialog.

Report Version

Specifies the report version.

Catalog Version

Specifies the catalog version.

Priority

Specifies a priority level to the report running task. The priority levels are from 1 to 10 in ascending order of lowest priority to highest priority. This property is available to administrators. By default this property is ignored unless server.properties is modified to set queue.policy not equal to 0.

Parameter tab

This tab allows you to specify the parameter values if any to run the report.

Enter Parameters

Displays parameters of the report with your last-time saved default values, which could be the values saved in this tab last time, or in the Enter Parameter Values dialog or Parameter Settings dialog, or when scheduling the report, and for a web report, could also be the values saved in the Parameters panel. Edit the values according to your requirement. You may specify parameter values in these ways.

If you have not yet set the default values on the server, or if you did but your saved default values cannot fully match the current parameters, all the parameters will use their default values specified in the parameters' definition as the initial values.

If no parameter is used in the report, "No Parameter Needed" will be displayed here.

Use Saved Values

If it is available, you can select the previously saved parameter values to apply to the report and save parameter values for reuse later.

Save as default

Saves current parameter values as the default values for the report. It is a user-report level setting.

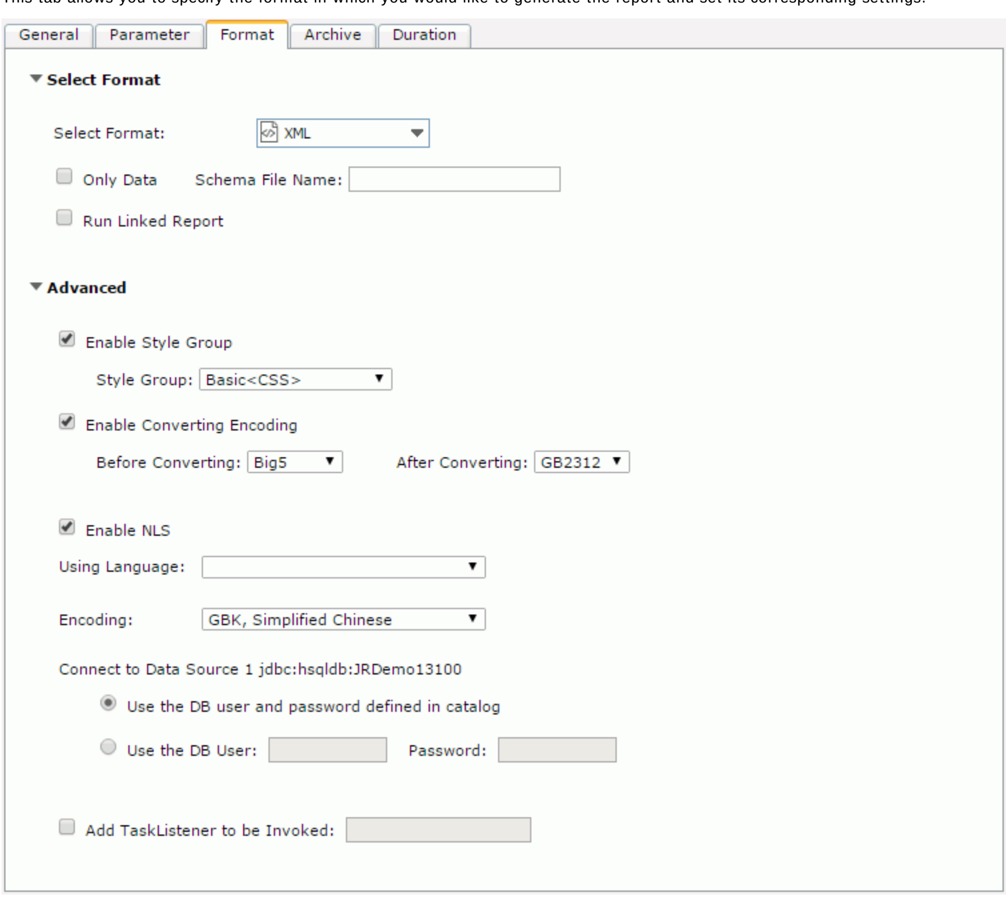
This option is an action and it takes effect after the task is submitted. Its initial status is always unselected.

For page reports, this option is available when Enable Setting Default Parameter Values For Page Report is checked in the Profile > Customize Server Preferences > Advanced tab.

For web reports, this option is available when Enable Setting Default Parameter Values For Web Report is checked in the Profile > Customize Server Preferences > Advanced tab.

Format tab

This tab allows you to specify the format in which you would like to generate the report and set its corresponding settings.



Select Format section

Specifies the format of the report result.

Select Format

Specifies the format in which the report result will be. It can be one of the following:

- Web Report
- Page Report
- HTML
- PDF
- Excel
- Text
- RTF
- XML
- PostScript
- Applet

Advanced section

Specifies some advanced format settings.

Enable Style Group

If a style group has been specified from the Override Style Group drop-down list in the Profile > Customize Server Preferences > Advanced tab, this option gets checked and the specified default style group is selected in the Style Group drop-down list by default. You can also choose another style group from the Style Group drop-down list to run the page report tab or web report.

When the <No Style> item is selected in the Override Style Group drop-down list, this option is unchecked by default and the style group property of the page report tab or web report for this format that is predefined in JReport Designer will be used to run the page report tab or web report.

Enable Converting Encoding

Specifies whether to enable the conversion of encoding. If checked, the Before Converting and After Converting options will be activated. Select the encoding from the drop-down lists as required.

Enable NLS

Specifies whether to enable NLS for the report. If checked, the Using Language drop-down list will be displayed for you to choose a language.

Note: If there is no NLS resource defined for the report, you can only run the report using the default language.

Encoding

Specifies the encoding for the report.

Connect to [Data Source Name] [Connection Name]

Specifies the DB user and password with which you want to connect to the data source the report uses. For the default connection, [Connection Name] will not be shown or recorded.

- **Use the DB user and password defined in catalog**
  If checked, the DB user and password defined in the catalog will be used.
- **Use the DB User**
  If checked, specify another DB user and password instead of the one defined in the catalog.

Add TaskListener to be Invoked

Specifies whether to enable the user to call the Java application before/after viewing. Enabled only when the format is not specified to Page Report Studio.

Web Report

Runs a web report in Web Report Studio.

Resolution

Specifies the resolution of the result to zoom in/out, in DPI. The default value is obtained from the operation system, which is the resolution of your monitor, for example, 72 DPI on Unix or 96 on Windows. You can set higher/lower value to zoom in/out.

Page Report

Runs a page report in Page Report Studio.

Resolution

Specifies the resolution of the result to zoom in/out, in DPI. The default value is obtained from the operation system, which is the resolution of your monitor, for example, 72 DPI on Unix or 96 on Windows. You can set higher/lower value to zoom in/out.

Profile

Specifies the Page Report Studio profile to be applied to run the report, which contains a set of Page Report Studio settings.

Background Mode Timeout

Specifies the time after which the report will continue to run in background mode. When a page report runs and the results have not yet been generated after the specified time, it will automatically be switched to run in background mode. Once the report is complete you can find the results in the My Tasks > Background Tasks tab.

HTML

No Margin

Removes the margins you originally set while designing the report.

Multiple Files

Generates the report result to multiple HTML files. JReport designates a serial number for each HTML page. For example, if you named a 3-page report as "sales", JReport will create three files called sales_1.html, sales_2.html, and sales_3.html.

- **Embedded CSS**
  Specifies to embed the cascading style sheet (CSS) in the HTML files; otherwise, the .css file will be generated individually.

Single File

Generates the report result to a single HTML file.

- **No Hyperlink**
  If checked, there will be no hyperlinks for navigating previous and next pages on the navigation bar of the generated HTML file.
- **No Page Number**
  If checked, there will be no page number information showing the current page number and total page number on the navigation bar of the generated HTML file.

Drilldown

Generates the report result into an HTML file with the Drilldown feature enabled. The Drilldown feature enables you to inspect certain items for further detailed data.

Section 508 Compliant Output

If checked, the accessibility attributes defined for the report elements via the Report Inspector will be generated to the HTML result which is Section 508 compliant. See Accessibility for more information.

When Section 508 Compliant Output is checked, the Use HTML Data Table and Relative Font Size options will be checked and disabled. The output will be Section 508 compliant including HTML data table, accessible attributes, and relative font feature.

Use HTML Data Table

Specifies whether the table and crosstab components will be output as table objects in the HTML result.

Absolute Font Size

Generates the report result using an absolute font size, which means that the font size is fixed, and cannot be adjusted according to the font size settings in the web browser.

Relative Font Size

Generates the report result using a relative font size. The font size can be adjusted according to the font size settings in the web browser.

Use Chart

When publishing to e-mail or to FTP, only "Image Chart" is available for customization and "Use Chart" and "Applet Chart" is not needed.

- **Applet Chart**
  Uses a Java applet to display the charts in an HTML result file.

  If checked, on Google Chrome, the report can run correctly only when NPAPI is enabled for the web browser.
- **Image Chart**
  If selected, charts will be displayed as images. You can specify the image type from the drop-down list. The options are:
  - **Auto-select**
    If selected, the image format will be detected to JPG or GIF by the JReport system automatically. If the image colors are less than 256 colors, GIF will be applied; otherwise, it is JPG.
  - **GIF**
    If selected, the GIF format will be applied, which is a lossless compression technique and supports only 256 colors. GIF is better than JPG for images with only a few distinct colors, such as line drawings, black and white images and small text that is only a few pixels high.
  - **JPG (JPEG)**
    If selected, the JPG (JPEG) format will be applied, which is supported on the Web. JPG is a lossy compression technique that is designed to compress color and grayscale continuous-tone images. JPG images support 16 million colors and are best suited for photographs and complex graphics.
  - **PNG**
    If selected, the PNG format will be applied, which provides a portable, legally unencumbered, well-compressed (effectively 100 percent lossless compression), well-specified standard for lossless bitmapped image file. PNG supports indexed-color images of up to 256 colors and shows a more interchangeable, flexible and robust function than GIF.

Resolution

Specifies the resolution of the HTML result to zoom in/out, in DPI. The default value is obtained from the operation system, which is the resolution of your monitor, for example, 72 DPI on Unix or 96 on Windows. You can set higher/lower value to zoom in/out.

Web Browser

Specifies the web browser for which the HTML result adapts.

Text Overflow

Specifies whether the text overflow is visible or hidden.

Run Linked Report

If the report is linked with other reports, you can check whether or not to generate the linked reports in the HTML result. If you are only interested in the primary report, leave this option unchecked. Generating linked reports at the same time, especially when the linked reports contain a large amount of data, will cause performance issue.

PDF

No Margin

Removes the margins you originally set while designing the report.

Compress Image

Compresses the images in the report by the percentage you specify in the box.

Generate charts and barcodes using images (recommended)

When you run a report in PDF format, JReport will take the result of the whole report as a graphic to transform the report by the method of simulated printer and generate the report result in PDF format.

Generate charts and barcodes using vector graphics

The result of using Generate charts and barcodes using vector graphics is the same as with using Generate charts and barcodes using images (recommended). However, when you run a report in PDF format using Generate charts and barcodes using vector graphics, JReport will take the result of the whole report as a dataset to transform the report by sequence and to generate the report result in PDF format.

TOC

Generates the report result to PDF format with a Table of Contents. Not supported for web report.

Drilldown

Generates the report result to a PDF file with the Drilldown feature enabled. The Drilldown feature enables you to inspect certain items for further detailed data.

Encrypt

Specifies whether to encrypt the PDF file. If checked, click the Settings button to configure the encrypt settings in the Encrypt dialog.

Sign

Specifies whether to add the digital sign to the PDF file. If checked, click the Settings button to configure the sign settings in the Sign dialog.

Run Linked Report

If the report is linked with other reports, you can check whether or not to generate the linked reports in the PDF result. If you are only interested in the primary report, leave this option unchecked. Generating linked reports at the same time, especially when the linked reports contain a large amount of data, will cause performance issue.

Excel

Excel Version

Specifies the Excel version to be used.

- **Excel 97-2003 Workbook (*.xls)**
  Runs the file as .xls format.
- **Excel Workbook (*.xlsx)**
  Runs the file as .xlsx format.

Format

Specifies the format of the Excel result.

- **Report Format**
  If checked, JReport will attempt to make the formatting of the report in Excel match the format as designed in the template. Usually this format is recommended if you just want to view the report in Excel.
- **Column Format**
  If checked, the calculation method used for all components' row/column values in the report will be decided by the Columned property value of the report specified at report design time when exporting. This method is recommended when you want to actively use the result in Excel such as filtering and sorting.
- **Data Format**
  If checked, only the report data will be generated without format. This option is only available for the Excel version Excel 97-2003 Workbook (*.xls).

More/Less Options

Click to show/hide the additional settings for running the report to Excel. When Data Format is selected, only Word Wrap is available here.

- **Include Shapes in Export**
  Specifies whether to include the drawing objects in the Excel result, such as line, oval, and box.
- **Print Page Header**
  Specifies the page header text for the printed file.
- **Print Page Footer**
  Specifies the page footer text for the printed file.
- **Word Wrap**
  Specifies the word-wrap settings.
  - **All Keep Existing**
    Keeps all the settings of each object's Word Wrap property originally specified in the report.
  - **All Disabled**
    Disables the Word Wrap property for all objects. That is, the Word Wrap property is made false for all objects.
  - **All Enabled**
    Enables the Word Wrap property for all objects. That is, the Word Wrap property is made true for all objects.
- **Print Gridlines**
  Specifies whether to include gridlines when printing the Excel result.
- **Run Linked Report**
  If the report is linked with other reports, you can check whether or not to generate the linked reports in the Excel result. If you are only interested in the primary report, leave this option unchecked. Generating linked reports at the same time, especially when the linked reports contain a large amount of data, will cause performance issue.

Text

Delimited Format

Generates the report result to a standard Text file, using a delimiter you specify to separate the fields.

- **Use Quote Mark**
  Specifies to use quote marks in the Text file.
- **Repeat Last Column Value If Null**
  If checked, when a cell in the generated CSV Text file has no value, value of the previous cell in the same column will be used.
- **Customize Delimited**
  Fields in the generated Text file will be separated by a user defined delimiter. You can type your own delimiter in the Delimiter box.
- **Tab Delimited**
  Specifies to use a Tab delimiter to separate the fields.
- **CSV Format**
  Exports to CSV format Text file.

Horizontal Density

Specifies the value for each unit of the horizontal density between columns. The resulting density is a direct ratio of the value you specify. That is, the greater the value, the smaller the width between columns. By default the density will be specified by JReport.

Vertical Density

Specifies the value for each unit of the vertical density between columns. The resulting density is a direct ratio of the value you specify. That is, the greater the value, the smaller the height between columns. By default the density will be specified by JReport.

Notes:

- When using user defined densities, if the densities are not set appropriately, the fields in the report may overlap each other, so you are not recommended to use this way to run the report result to Text.
- When setting the value of Horizontal/Vertical Density, you need to pay attention to the following:
  - The value of Horizontal/Vertical Density must be greater than the character's width/height of the smallest field in the report (smallest field is the field with the smallest font size), otherwise, the value you set will not be applied.
  - If the value of Vertical Density is greater than 0 and the value of Horizontal Density is less than 0, the value that you specify for the Vertical Density will be applied and the value of Horizontal Density will be specified by JReport.
  - If the value of Vertical Density is less than 0 and the value of Horizontal Density is greater than 0, the value that you specify for the Horizontal Density will be applied and the value of Vertical Density will be specified by JReport.
  - If the values of Vertical Density and Horizontal Density are both greater than 0 and the value of Horizontal Density is less than 11, the specified value of the two densities will be applied. Otherwise, they will be specified by JReport.
  - If the values of Vertical Density and Horizontal Density are both less than 1, these densities will be specified by JReport.

Compress

Specifies whether to generate the report result to Text format in a compressed size, that is to say, there will be no clearance between the columns.

Header and Footer

If checked, the Text file will contain all headers and footers in the report, including Report Header/Footer, Page Header/Footer and Group Header/Footer. Otherwise, the Text file will only contain data in Detail panel.

Windows End-of-line (CR-LF)

Specifies to use Windows end-of-line characters to indicate the start of a new line. If checked, two characters <cr> and <lf> will be used at the end of the line.

Unix End-of-line (LF)

Specifies to use Unix End-of-line characters to indicate the start of a new line. If checked, only the Unix End-of-line character <lf> will be used.

Run Linked Report

If the report is linked with other reports, you can check whether or not to generate the linked reports in the Text result. If you are only interested in the primary report, leave this option unchecked. Generating linked reports at the same time, especially when the linked reports contain a large amount of data, will cause performance issue.

RTF

Best Editing

Specifies whether to apply flow layout for the RTF result.

No Margin

Removes the margins you originally set while designing the report.

Run Linked Report

If the report is linked with other reports, you can check whether or not to generate the linked reports in the RTF result. If you are only interested in the primary report, leave this option unchecked. Generating linked reports at the same time, especially when the linked reports contain a large amount of data, will cause performance issue.

XML

Only Data

- If checked, the generated XML file will only contain the database column information. The generated XML schema file will only contain the structure information of the report.
- If unchecked, the generated XML file will also contain elements controlled by formulas, and the generated XML schema file will contain all the detailed information from the report, including all the property values of each report object.

Schema File Name

Specifies the directory and the name of an existing XML schema (.xsd) file. If you specify an existing XML schema file, the generated XML file will be based on it. Otherwise, a new XML schema file will be generated to the directory where the XML file is generated. The new XML schema file and the XML file will have the same name but with different extensions.

Run Linked Report

If the report is linked with other reports, you can check whether or not to generate the linked reports in the XML result. If you are only interested in the primary report, leave this option unchecked. Generating linked reports at the same time, especially when the linked reports contain a large amount of data, will cause performance issue.

PostScript

No Margin

Removes the margins you originally set while designing the report.

Run Linked Report

If the report is linked with other reports, you can check whether or not to generate the linked reports in the PostScript result. If you are only interested in the primary report, leave this option unchecked. Generating linked reports at the same time, especially when the linked reports contain a large amount of data, will cause performance issue.

Applet

Pop-up Window

Specifies to show an Applet window when viewing a report.

Zip Result

Specifies to compress the result and its size would be smaller.

Java Plug-In 1.2 for Windows

Directs applets to run using Sun's Java Runtime Environment (JRE).

Java Plug-In 1.3 for Windows

Directs applets to run using Sun's Java Runtime Environment (JRE).

Run Linked Report

If the report is linked with other reports, you can check whether or not to generate the linked reports in the Applet result. If you are only interested in the primary report, leave this option unchecked. Generating linked reports at the same time, especially when the linked reports contain a large amount of data, will cause performance issue.

Notes:

- You'd better specify a file destination when running a report via Applet, such as `D:\folder\filename.pdf` in case you may not find the result. By default when running a report via Applet on Internet Explorer, the generated result will be located on the Desktop, and when on Firefox, it will be in its installation root.

  If you do not have read or write permission to the default location or the location you specified, you will get an access denied error. To handle this, add the required permission in jdk\jre\lib\security\java.policy that the applet uses:

  For the default location "Desktop":

  ```
  permission java.util.PropertyPermission "user.dir", "read";
  permission java.io.FilePermission "${user.home}${/}Desktop${/}*", "read,write";
  ```

  For user defined location:

  ```
  permission java.io.FilePermission "D:${/}temp${/}*", "read,write";
  ```

- When advanced running a report to Applet on Firefox, after you select File > Open Result File on the Report Viewer, you may get the access to the Open dialog denied error as follows:

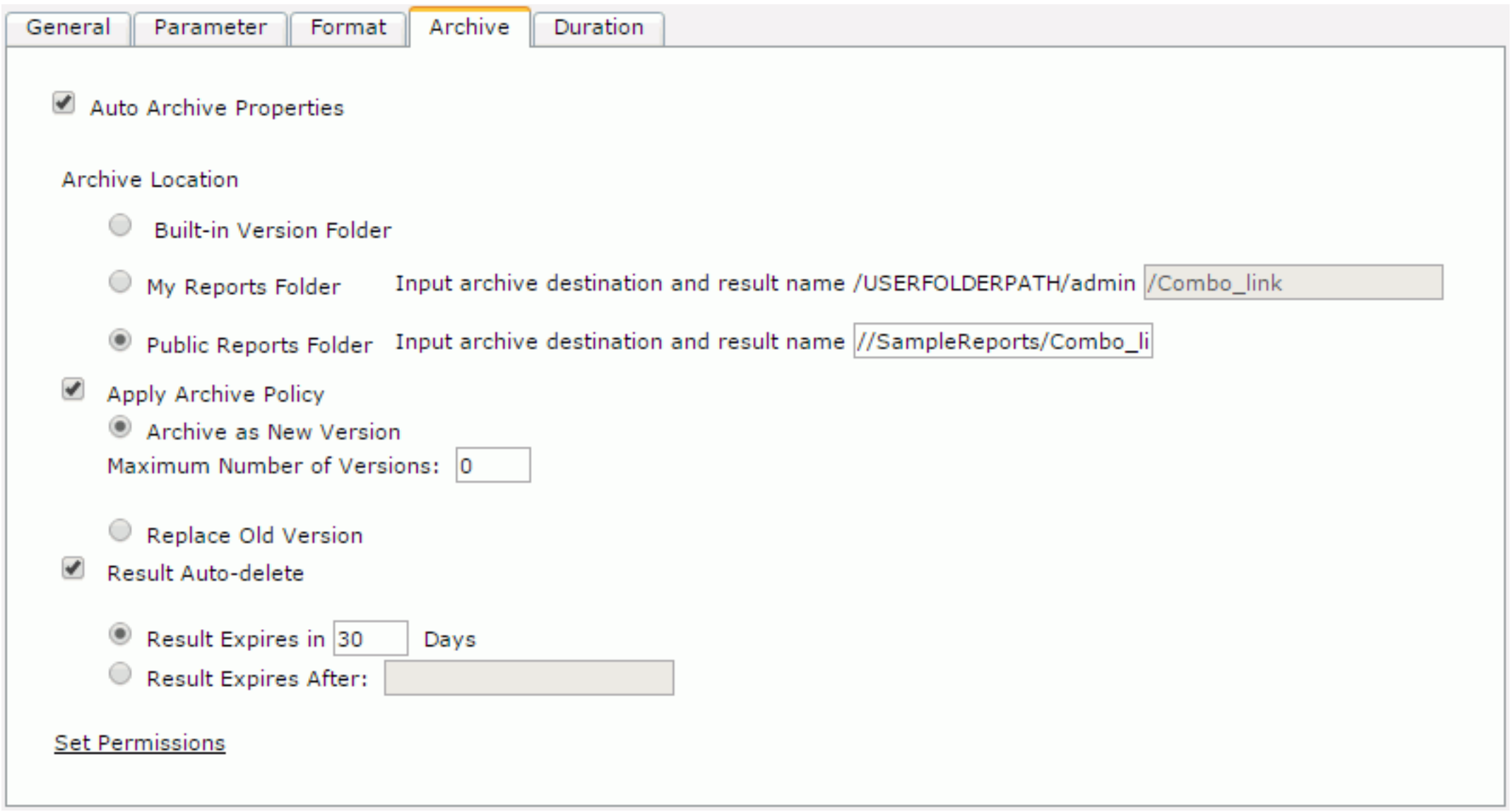
  ```
  java.security.AccessControlException: access denied (java.io.FilePermission Open read)
  ```

  To resolve this, add the required permission in java.policy that the applet uses:

  ```
  permission java.io.FilePermission "Open", "read,write";
  ```

Archive tab

Options below are available only when the view format is NOT set as Page Report or Web Report in the Format tab.



Auto Archive Properties

Specifies to archive the result version automatically after the report has finished running.

Archive Location

Specifies the location in which to archive the report result version.

- **Built-in Version Folder**
  Specifies to save the report result version to the built-in version folder. When the organization feature is enabled and the report is from the Public Reports folder, this option is not available to organization users.
- **My Reports Folder**
  Specifies to save the report result version to the My Reports folder.
- **Public Reports Folder**
  Specifies to save the report result version to the Public Reports folder. Not available to organization users.
- **Organization Reports Folder**
  Specifies to save the report result version to the Organization Reports folder. Available to organization users only.

Input archive destination and result name

Specifies the name and location with which to archive the result.

Apply Archive Policy

Applies an archive policy to the report result version.

- **Archive as New Version**
  Specifies whether to use multiple versions for the report result.
  - **Maximum Number of Versions**
    Specifies the maximum number of versions that will be listed in the version table of the report result. The default value is 0, which means that the version number is unlimited.
- **Replace Old Version**
  Specifies to replace the old version when a new version is generated.

Result Auto-delete

Specifies the deleted date of the report result version. You should make sure to set the result version as deleted automatically within one hundred years. If the time you specified exceeds one hundred years, JReport Server will keep the result forever.

- **Result Expires in N Days**
  Specifies a period after which the report result version will be automatically deleted.
- **Result Expires After**
  Specifies a certain day on which the report result version will be automatically deleted.

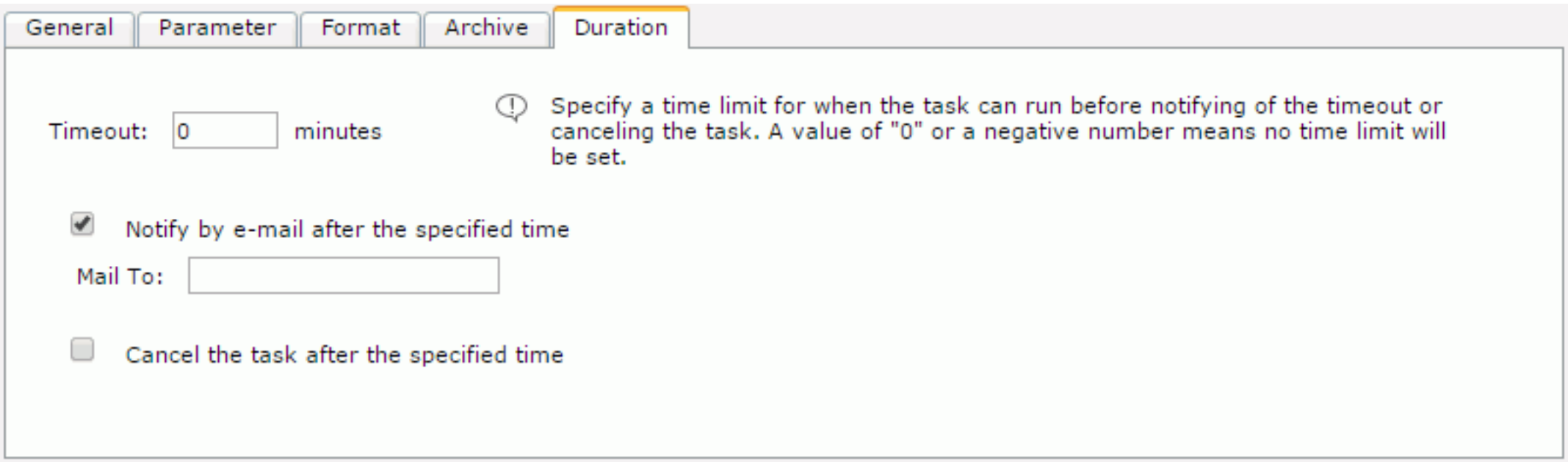
Set Permissions

Available only when the Archive Location is specified to be Public Reports Folder. Click the link to set user permissions to the specified report in the Set Permissions dialog.

Duration tab

This tab is available only when you have enabled the task-level timeout mechanism either by setting the task.duration.enable property to true in the server.properties file or by checking the Enable Task Duration option on the JReport Administration page > Configuration > Advanced panel.

In this tab, you can specify a time duration for a task, and ask JReport Server to cancel the task or to notify you or someone else of the task status via e-mail if the task has not yet finished running when the task duration is up.



The following are the available options:

Timeout

Specifies the time duration for the task.

Notify by e-mail after the specified time

Sends an e-mail about the task information when the specified time is up.

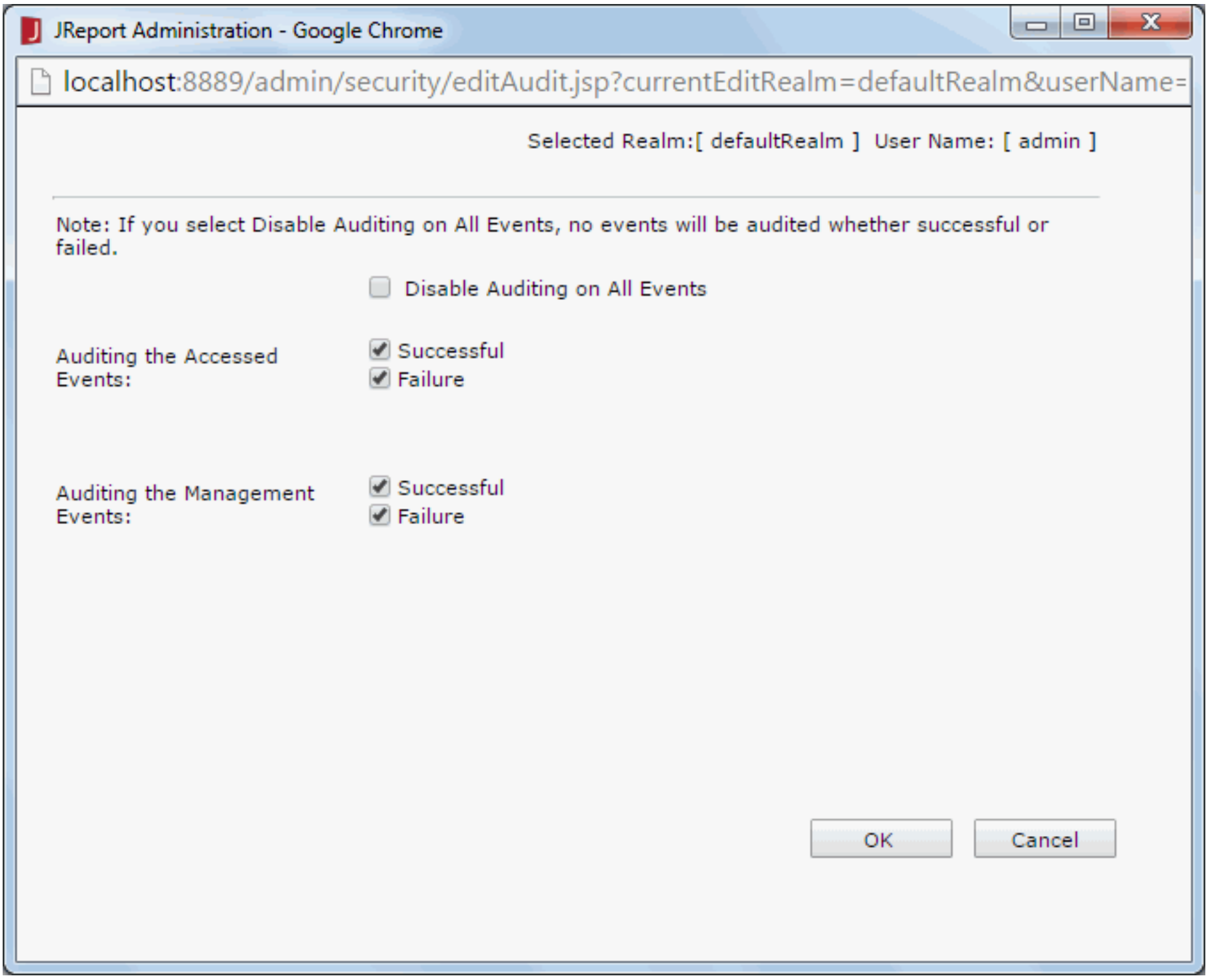
- **Mail To**
  Specifies the e-mail address of the recipient.

Cancel the task after the specified time

Cancels the running task when the specified time is up.

Auditing dialog

This dialog appears when you click the Auditing link of a user in the User panel of the Security dialog on the JReport Administration page.



Disable Auditing on All Events

Specifies whether to audit on all events.

Auditing the Accessed Events

Specifies to audit accessed events and record them in the log file in `<install_root>\logs`.

- **Success**
  Specifies to audit and record the accessed events when they have been successfully performed.
- **Failure**
  Specifies to audit and record the accessed events when they fail.

Auditing the Management Events

Specifies to audit the management events and record them in the log file in `<install_root>\logs`.

- **Success**
  Specifies to audit and record management events when they have been successfully performed.
- **Failure**
  Specifies to audit and record management events when they fail.

OK

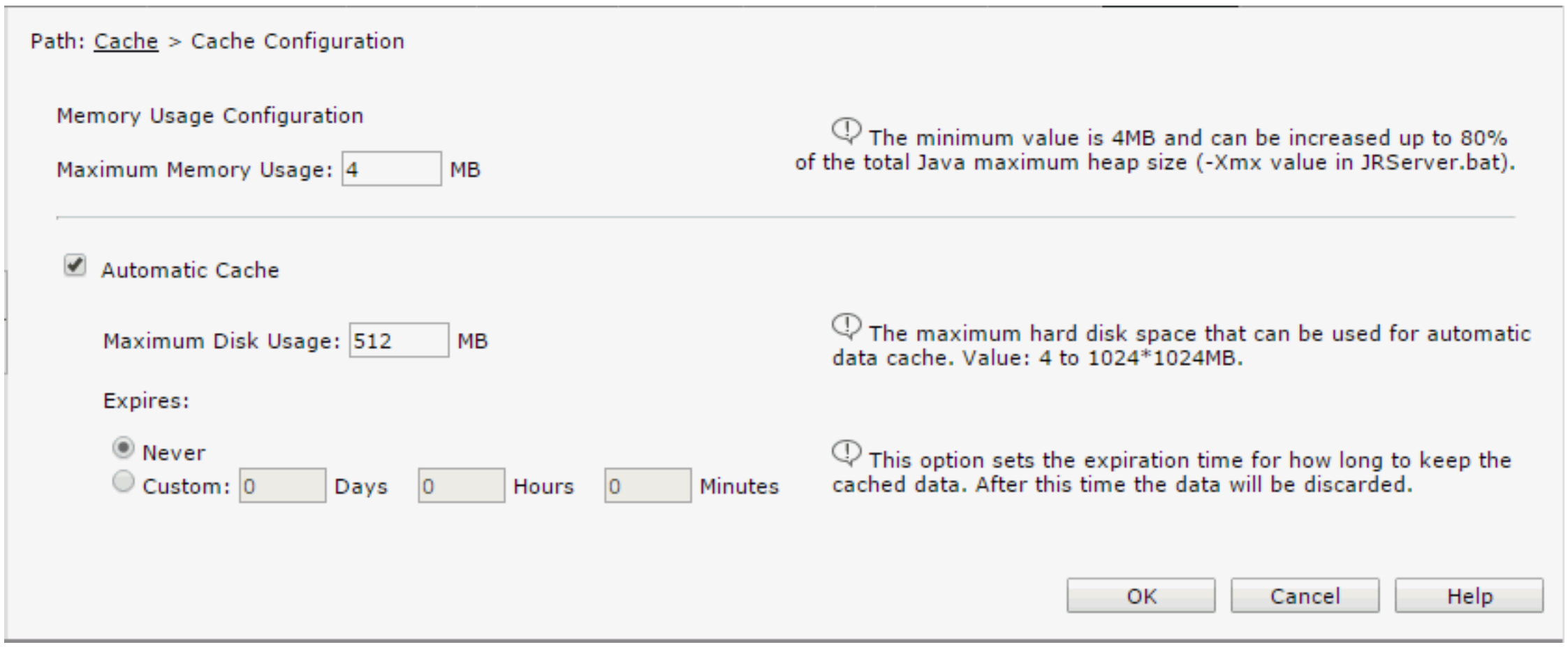
Audits and records the events with the information you have specified.

Cancel

Cancels operations and closes the dialog.

Cache Configuration dialog

The dialog appears after you click the Cache Configuration link on the JReport Administration page > Cache > Data Cache > Cached Report Data tab. It allows you to configure memory usage for cached report data (CRD) and configure auto cached report data.



Memory Usage Configuration

Configures the memory usage.

- **Maximum Memory Usage**
  Specifies the maximum memory usage when running a report using the CRD. The minimum value is 4 MB and can be increased up to 80% of the total Java maximum heap size (-Xmx value in JRServer.bat). All CRDs being used to run reports will share the same memory space. If possible, set the maximum memory usage to a number that the memory can tolerate, for example, 512 MB or bigger.

Automatic Cache

Specifies whether to enable auto CRD. When it is selected, the following options are available:

- **Maximum Disk Usage**
  Specifies the maximum hard disk space for auto CRD. The value should be between 4 MB and 1024*1024 MB.
- **Expires**
  Specifies how long an automatic cached sharable data result is kept.
  - **Never**
    The auto CRD will not expire.
  - **Custom**
    Specifies the period an auto CRD exists.

OK

Applies the changes and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Help

Displays the help document about this feature.

Related topics:

- Managing cached report data

Cache dialog

After you click Cache on the system toolbar of the JReport Administration page, the entries to the following panels are available on the drop-down menu. The panels help you to configure the cache settings of JReport Server.

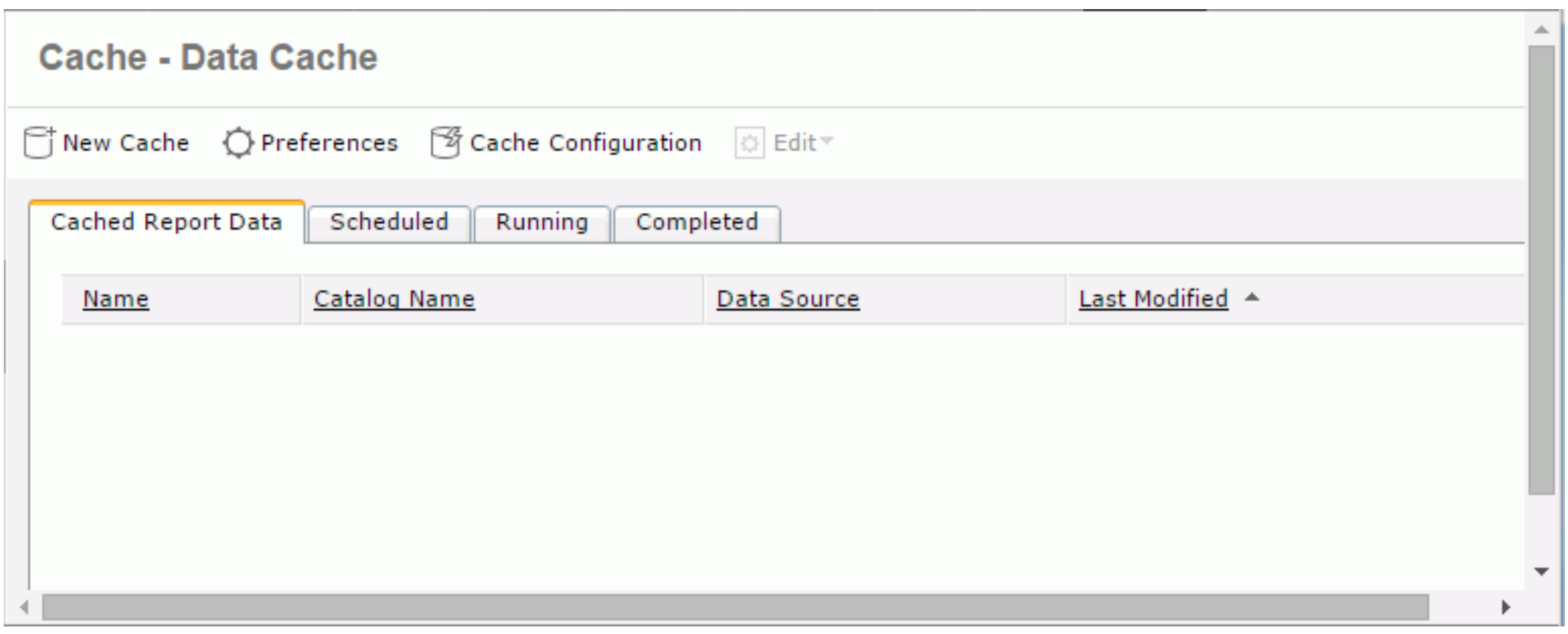
The JReport data cache system allows you to schedule and cache just the datasets used in your reports without actually running the reports. When the end user selects a report that is linked to a cached result set the data is retrieved from the cache and not from the DBMS. This is especially useful for month and quarter end reports where multiple reports use the same dataset with the same parameters such as begin and end date. See Managing cached report data for more information.

The report cache allows report templates and catalogs to be cached for faster access. The security cache allows you to configure how many users, groups and roles to cache without going back to the JReport Server database. The image cache allows you to configure how many images to cache.

- Data Cache panel
- Report Cache panel
- Security Cache panel
- Image Cache panel

Data Cache panel

The panel is displayed after you click Cache > Data Cache on the system toolbar of the JReport Administration page. It allows you to set up caching report data.



The panel contains the following tabs:

- Cached Report Data
- Scheduled
- Running
- Completed

The following are options available on the task bar when different tab is selected:

New Cache

Opens the New Cache dialog in which you can schedule a task that defines when to cache data of the specific query. Available when Cached Report Data tab is selected.

Preferences

Specifies which columns will be displayed in the current tab.

Cache Configuration

Opens the Cache Configuration dialog for setting the cache size and auto cache. Available when Cached Report Data tab is selected.

This option is not available to organization admin.

Edit

Available when Cached Report Data tab is selected.

- **Properties**
  Edits the scheduling information of the selected data cache in the tab.
- **Delete**
  Removes the selected data cache from the tab.

Delete

Removes the selected data cache task. Available when Completed tab is selected.

Cached Report Data

This tab displays the scheduled data cache tasks and consists of the following columns. In this tab, you can schedule tasks for generating a data cache, edit the scheduled tasks, and configure cache size and auto cache.

Column	Description
Name	The name of the scheduled data cache.
Path	The path of the scheduled data cache in the server resource tree.
Catalog Name	The catalog that the scheduled data cache belongs to.
Data Source	The data source that the scheduled data cache belongs to.
Data Type	The type of the query resource that the scheduled data cache is based on.
Last Modified	The last time when the scheduled data cache was modified.

Scheduled

This tab displays the data cache tasks that are waiting to be performed. It consists of the following columns.

Column	Description
Name	The name of the scheduled data cache.
Path	The path of the scheduled data cache in the server resource tree.
Catalog Name	The catalog that the scheduled data cache belongs to.
Data Source	The data source that the scheduled data cache belongs to.
Data Type	The type of the query resource that the scheduled data cache is based on.
Last Modified	The latest time when the task was modified.
Next Run Time	The next scheduled time this task is to be performed.
Last Run Time	The last scheduled time this task was performed.
Is Successful	Shows whether or not the last running of this task was successfully performed. The value true means that the last running was performed successfully and false means the task failed. If the column is empty, the task has not been run before.

Running

This tab displays the data cache tasks that are currently being performed. It consists of the following columns.

Column	Description
Name	The name of the scheduled data cache.
Path	The path of the scheduled data cache in the server resource tree.
Catalog Name	The catalog that the scheduled data cache belongs to.
Data Source	The data source that the scheduled data cache belongs to.
Data Type	The type of the query resource that the scheduled data cache is based on.
Start Time	The time when this task was started.
Parameters	The parameters of the query.

Completed

This tab displays the data cache tasks that have already been performed and consists of the following columns. In this tab, you can remove the record of a completed CRD task if required.

Column	Description
Name	The name of the scheduled data cache.
Path	The path of the scheduled data cache in the server resource tree.
Catalog Name	The catalog that the scheduled data cache belongs to.
Data Source	The data source that the scheduled data cache belongs to.
Data Type	The type of the query resource that the scheduled data cache is based on.
Is Successful	Shows whether this task was successfully performed. The value true means that the task was performed successfully and false that the task failed.
Completed Time	The time when this task was completed.
Parameters	The parameters of the query.

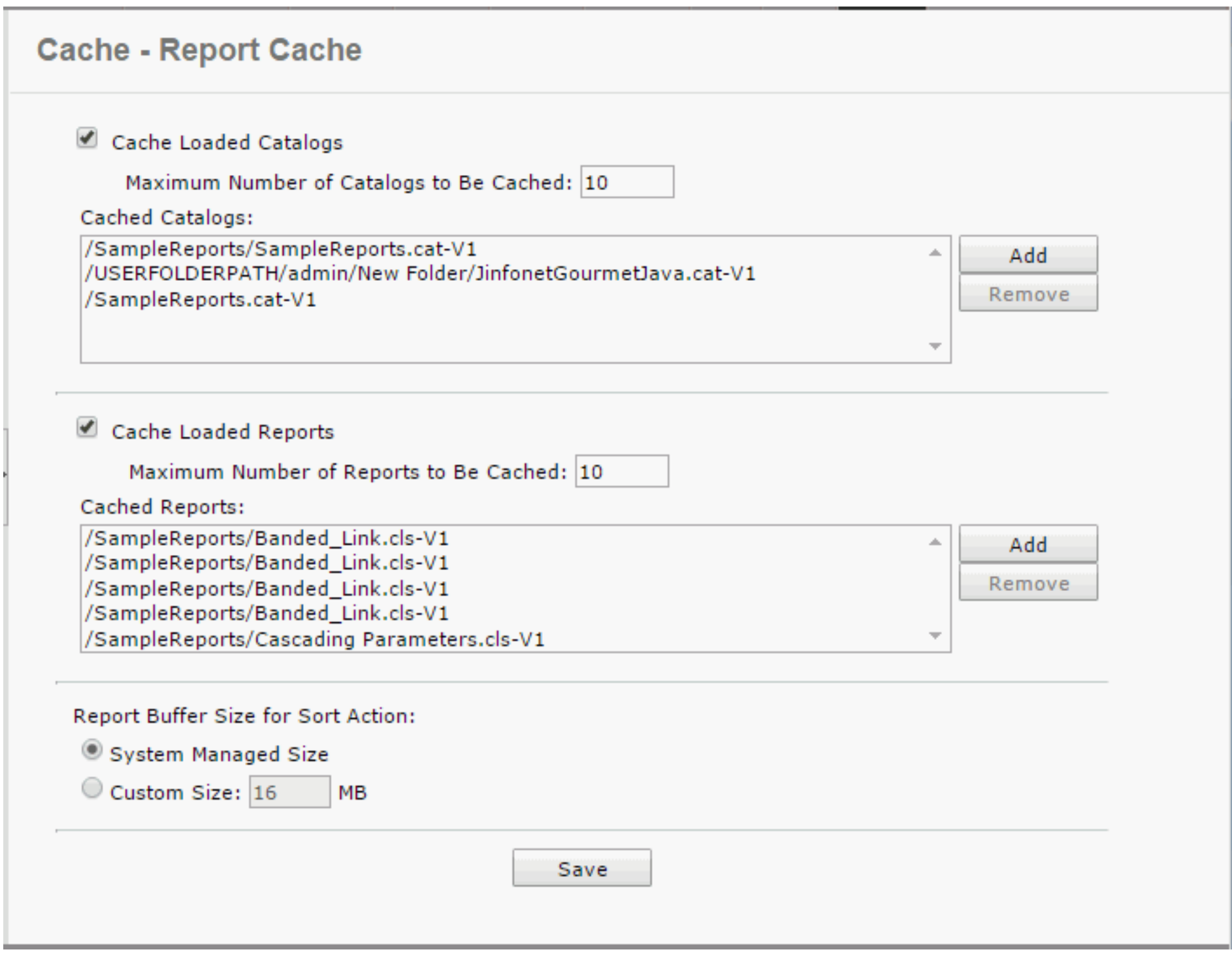
Related topics:

- Managing cached report data

Report Cache panel

The panel is displayed after you click Cache > Report Cache on the system toolbar of the JReport Administration page.

To improve performance, JReport Server enables you to cache reports and catalogs into memory so that they do not have to be loaded from disk when they are required. Note that when you run a report in Page Report Studio, the report and the catalog will not be cached.



Cache Loaded Catalogs

Specifies whether to keep a catalog in memory or to remove it from memory after a report has been generated.

Usually, after a report has been generated, the catalog used to generate it is removed from memory. However, if you specify this option, the catalog will be cached instead of being removed.

- **Maximum Number of Catalogs to Be Cached**
  Specifies the maximum number of catalogs that can be cached by the server in its memory. No more catalogs can be cached if this amount has been reached.

- **Cached Catalogs**
  Lists the currently cached catalogs.

  If you are using the Add button to add a catalog, the listed catalog item includes information such as catalog file path in the server resource tree and catalog version number. The format is as follows: `<catalog file path>-V<version number>`. The root path stand for the My Reports folder is `/USERFOLDERPATH/admin/`; the root path stand for the Public Reports folder is `/`.

  For the catalogs that are automatically cached by the server after the reports in them are run, the real paths of them are listed.

  - **Add**
    Adds a new catalog to the list of catalogs to be cached.

  - **Remove**
    Deletes the selected cached catalog.

Cache Loaded Reports

Specifies whether to keep the reports in memory or to remove them from memory after they have been generated.

- **Maximum Number of Reports to Be Cached**
  Specifies the maximum number of reports that can be cached in the memory by the server. No more reports can be cached if this amount has been reached.

  Each cached report will use some heap memory. If the heap is used up, an OutOfMemoryException will be thrown. In JRServer.bat/sh the -Xms setting specifies the initial heap and -Xmx specifies the maximum heap. For example, -Xms256m -Xmx512m means that the initial heap is 256MB and the maximum heap is 512MB. You can refer to the documents of the Java VM for more information.

  Here is an example, the accuracy depends on your OS and VM:

Cache Size	Heap Size Required
10 reports	16M
30 reports	25M

  The default cache size is 10 reports and the default maximum heap space is 512MB. When you adjust the number, you should adjust the heap accordingly. The larger the heap space, the better the performance provided enough physical memory is available.

- **Cached Reports**
  Lists currently cached reports.

  If you are using the Add button to add a report, the listed report item includes information such as report file path in the server resource tree, report version number, and applied style name. The format is as follows: `<report file path>-V<version number>(<applied style name>)`. When there is no style applied, the applied style name is None. The root path stand for the My Reports folder is `/USERFOLDERPATH/admin/`; the root path stand for the Public Reports folder is `/`.

  For the reports that are automatically cached by the server after they are run, the real paths of them are listed.

  - **Add**
    Adds new reports to the list to be cached.

  - **Remove**
    Deletes the selected cached reports.

Report Buffer Size for Sort Action

This option is not available to organization admin.

- **System Managed Size**
  Specifies whether to automatically control the buffer size used in the RAM for each report when sorting on page reports by system. If checked, all data for sorting will be stored in the RAM automatically.

- **Custom Size**
  Specifies to customize the buffer size used in the RAM for each report when sorting on page reports. The default value is 16 MB, and the minimum size required by JReport is 4 MB. Any value smaller than 4 MB will be ignored and use 4 MB instead, because if the amount of data for sorting is large (such as 5,000,000 records), but the buffer size that you configure is small, it will require a lot of IO and produce poor performance.

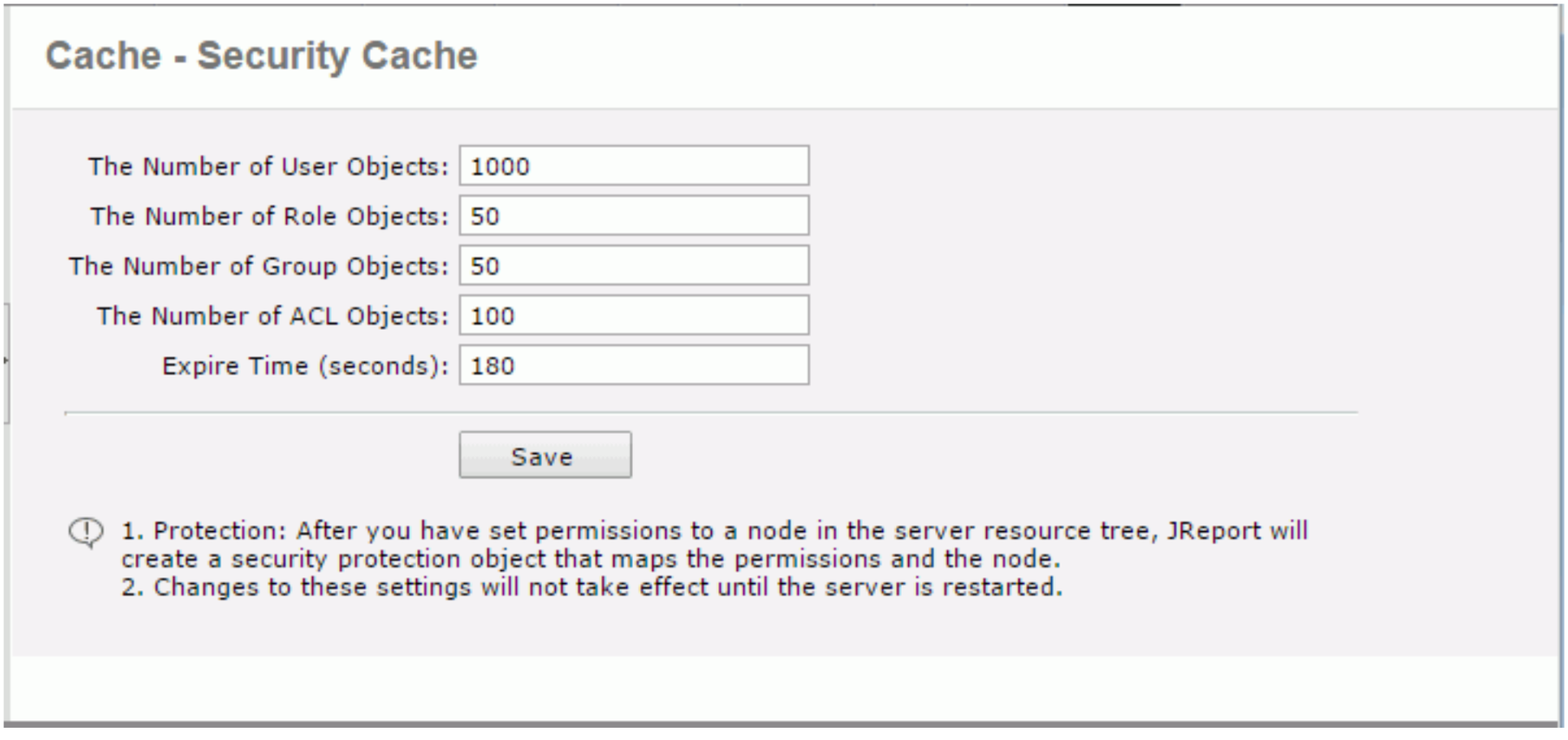
Note: Any change to the buffer size will take effect immediately.

Save

Applies the changes to JReport Server.

Security Cache panel

The panel is displayed after you click Cache > Security Cache on the system toolbar of the JReport Administration page. It allows you to cache security objects.



The Number of User Objects

Specifies the maximum number of user objects in the security cache.

The Number of Role Objects

Specifies the maximum number of role objects in the security cache.

The Number of Group Objects

Specifies the maximum number of group objects in the security cache.

The Number of ACL Objects

Specifies the maximum number of ACL objects in the security cache.

Expire Time

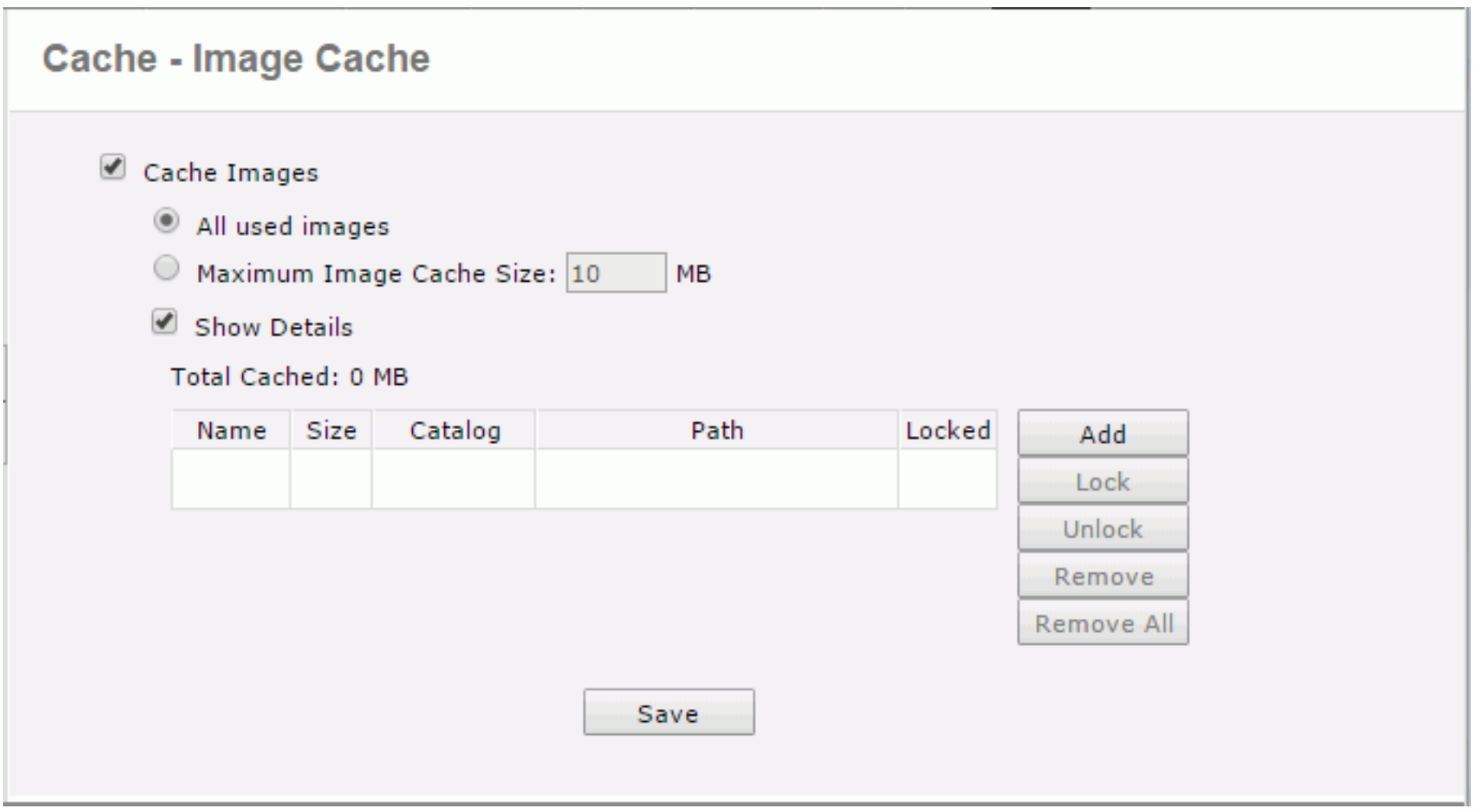
Specifies how long the security cache will be kept for. The time is measured in seconds.

Save

Applies the changes to JReport Server.

Image Cache panel

The panel is displayed after you click Cache > Image Cache on the system toolbar of the JReport Administration page. It allows you to cache images used in reports and dashboards and view the cached images.



Cache Images

Specifies whether or not to enable image cache.

- **All used images**
  If checked, all the images that are used in reports and dashboards will be cached.
- **Maximum Image Cache Size**
  If checked, specify the maximum size for caching images as required. The default value is 10 MB.
- **Show Details**
  If checked, the detailed status of the current image cache will be shown, that is to say, you can see all the cached images.
  - **Name**
    Displays the name of the image.
  - **Size**
    Displays the size of the image file.
  - **Catalog**
    Displays the catalog file in which the image file is.
  - **Path**
    Displays the path of the cached image.
  - **Locked**
    Displays the lock status of the image file.
  - **Add**
    Adds an image to the image cache.
    - **Select a Folder**

      Specifies the folder in the server resource tree that contains the required image. You can click the button ... to open the Select Folder dialog for selecting the folder.
    - **Catalog**
      Specifies the catalog that contains the required image. Appears after you have selected a folder first.
    - **Image File**
      Specifies the image file. Appears after you have selected a folder first.
  - **Lock**
    If you select one image and lock it, the image will be always kept in cache until you remove it manually or restart JReport Server.
  - **Unlock**
    The unlocked image will not be always be kept in cache.
  - **Remove**
    Removes the selected image from the image cache.
  - **Remove All**
    Removes all the images from the image cache.

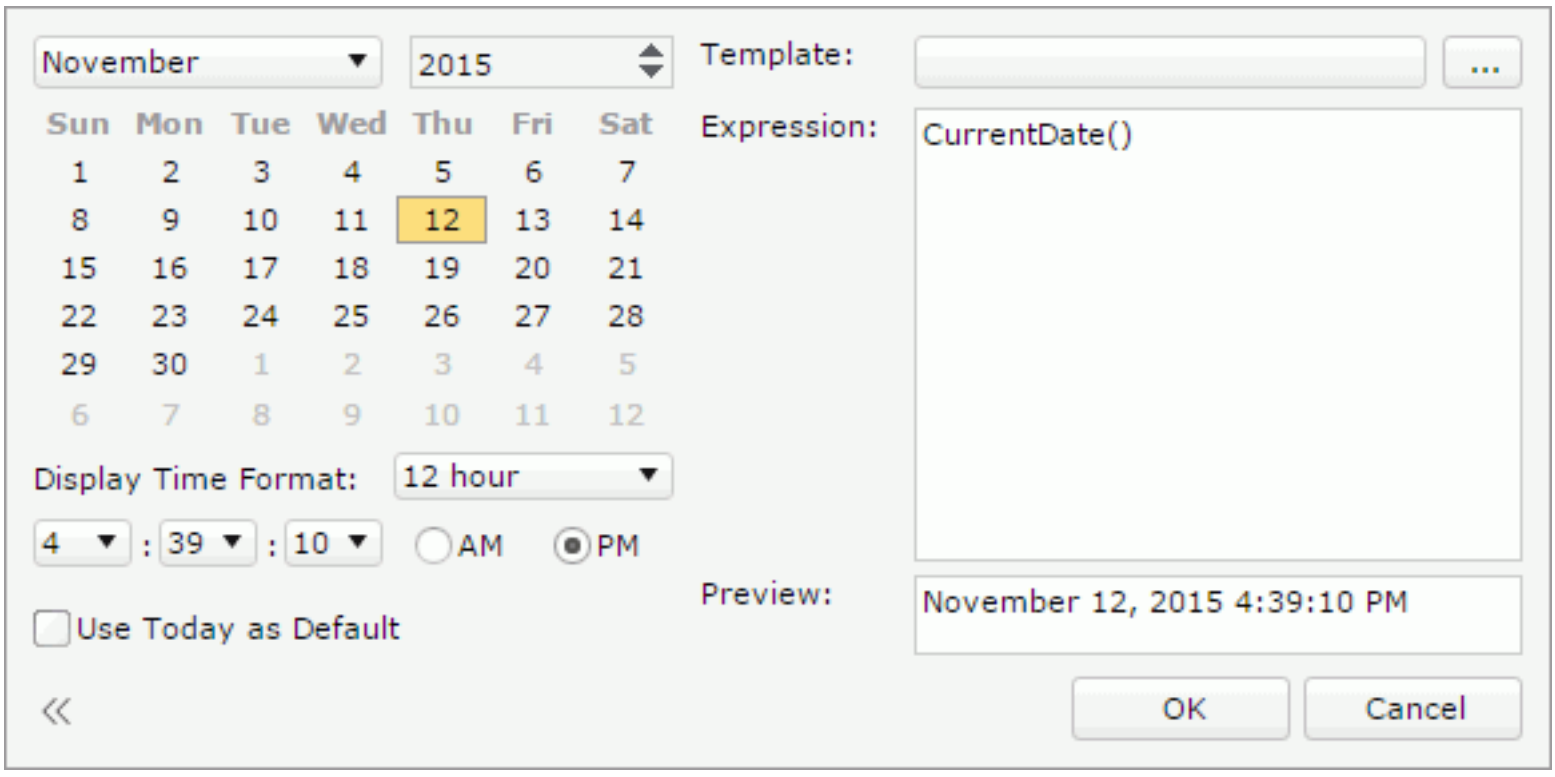
Save

Applies the changes to JReport Server.

Calendar dialog

The dialog appears when you click the calendar button while specifying value to a Date, DateTime, or Time parameter which allows for type-in values. It helps you to create a date and time parameter value using either calendar or expression.

The dialog is divided into two sections: the left section is a calendar for selecting date and time, the right section is for editing an expression by using built-in formula functions. The expression has higher priority than the value specified in the calendar. If you cannot see the expression settings, click **>>** at the bottom left corner.



Calendar

Specifies the month, year, day and time from the calendar. The time setting is available to DateTime or Time format parameters. The date/time in the calendar can be synchronized with the date/time calculated by a valid expression.

Display Time Format

Specfies in which format to display the time, 12-hour or 24-hour. Each time when the calendar is accessed, the default selected format is always 12-hour format.

Use Today as Default

This option is used to set the default selected date in the calendar. When it is checked, the calendar always shows the current date when it is accessed. When this option is unchecked, the default is the same as the value shown in the parameter's value box.

>>/<<

Displays or hides the expression settings.

Template

You can choose an existing expression template and then either use it directly or edit your expression based on it. The Template text field does not support typing in, if you would like to create an expression all by yourself, select the first blank row from the drop-down list.

Expression

Displays the expression. You can edit the expression directly in the text box.

Lists the built-in Date/Time formula functions. Single click to select a function and insert it into the Expression text box.

Preview

Displays the value result based on the specified expression, or gives error information if the expression is not correct.

OK

Sets either the value selected in the calendar or the expression as the parameter value. Expression has higher priority.

Cancel

Does not retain any changes and closes the dialog.

Related topics:

- Using built-in functions to set date and time parameter values

Catalog Properties dialog

You can set the properties of a specified catalog in its Properties dialog according to your requirements.

The Properties dialog is divided into the following two tabs:

- General
- Permission

Reset

Discards your modifications and restores the dialog to its default status.

OK

Retains the settings and submits the task to the server.

Cancel

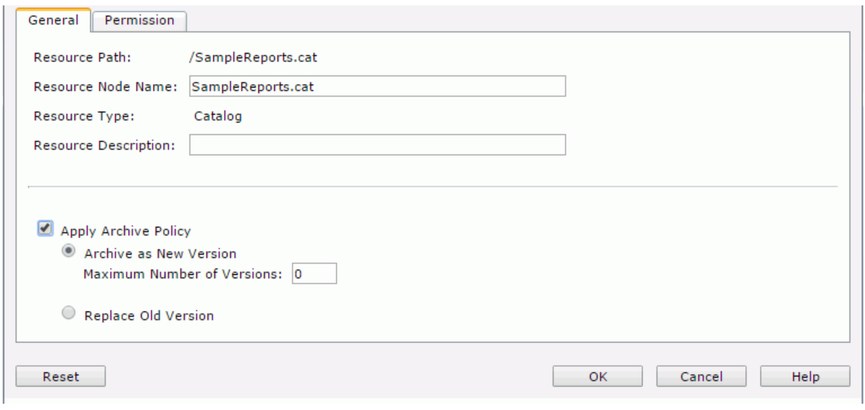
Cancels any settings and closes the dialog.

Help

Displays the help document about this feature.

General tab

Specifies the general properties of the catalog.



Resource Path

Shows the resource path.

Resource Node Name

Specifies the name for the catalog.

Resource Type

Shows the type of the resource.

Resource Description

Specifies the description of the catalog.

[Custom Field Name]

Specifies value of the custom field for the catalog. A custom field can be regarded as a resource property and is available when it is enabled.

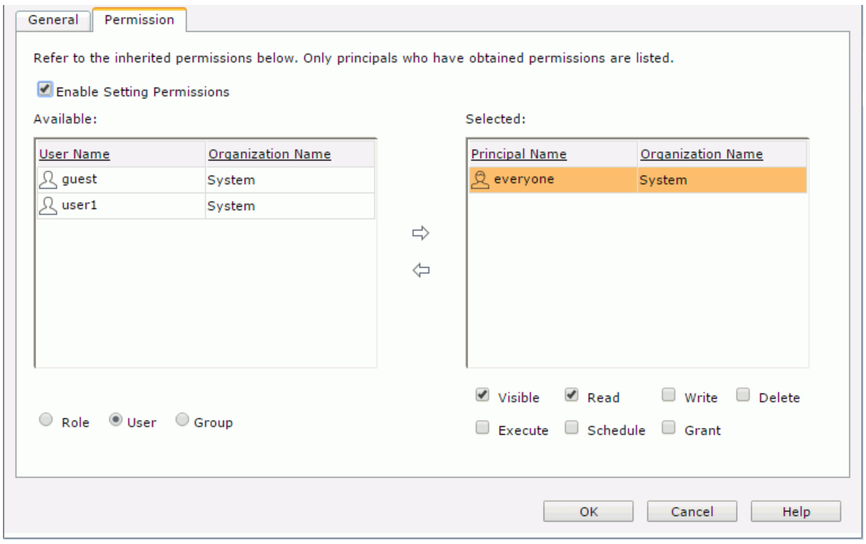
Apply Archive Policy

Applies an archive policy to the catalog versions.

- **Archive as New Version**
  Specifies whether to use multiple versions for the catalog.
  - **Maximum Number of Versions**
    Specifies the maximum number of versions that will be listed in the version table of the catalog. The default value is 0, which means that the version number is unlimited.
- **Replace Old Version**
  Specifies to replace the old version when a new version is generated.

Permission tab

Specifies permissions of roles/users/groups on the catalog. This tab is available when the catalog is in the Public Reports or Public Components folder.



Enable Setting Permissions

Enables the setting of permissions.

Available

Lists the roles/users/groups to which you can assign permissions. When organization is enabled, the Organization Name column will be available in the table.

- **Role**
  If checked, all roles will be displayed in the Available box for you to assign permissions.
- **User**
  If checked, all users will be displayed in the Available box for you to assign permissions.
- **Group**
  If checked, all groups will be displayed in the Available box for you to assign permissions.

Adds the selected role, user or group to the Selected box.

Removes the selected role, user or group from the Selected box.

Selected

Select a role/user/group in the Selected box and then select the permissions you would like the role/user/group to have on the catalog. For what each permission is used for, see Permission.

When organization is enabled, the Organization Name column will be available in the table.

Related topics:

- Changing resource and folder properties

Cluster dialog

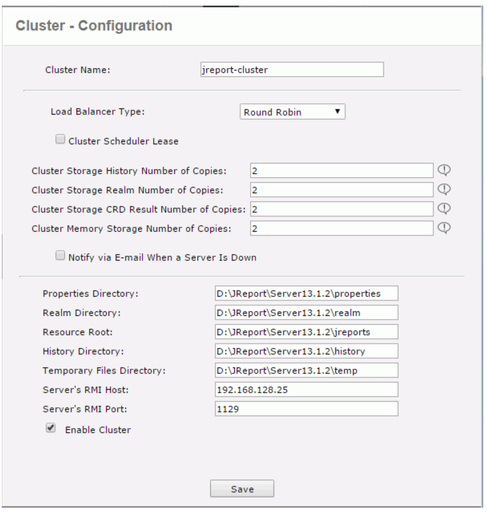
This dialog is displayed when you click Cluster on the system toolbar of the JReport Administration page. It helps you to configure the settings for a cluster. It is available only when your JReport Server license permits the cluster feature.

There are two panels in this dialog:

- Configuration panel
- Weight panel

Configuration panel

Use this panel to configure the servers in a cluster.



Cluster Name

Specifies a name for the cluster. When cluster is enabled, you must specify a unique cluster name for all servers that will join the cluster to build the cluster.

Load Balancer Type

Specifies the built-in load balancing algorithm. Configurable algorithms for load balancing clustered servers are: Min-load, Round Robin, Weighted Min-load, and Random. For details about how these algorithms work, see JReport Server Cluster main features.

Cluster Scheduler Lease

Specifies whether to enable lease for a scheduler in the cluster. If this option is left unchecked, every scheduler in the cluster becomes an active scheduler.

Lease is a semaphore that enables a scheduler to be an active scheduler. If a scheduler is enabled with lease, it becomes an active scheduler and will compete to trigger the schedule. Each active scheduler can hold a lease for a period of time (which depends on the value you set in the Cluster Scheduler Lease Valid Time text field), and then transfers it to another scheduler in the cluster.

- **Cluster Scheduler Lease Active Count**
  Specifies the active lease number in the cluster. By setting the active lease number, you get to know how many active schedulers there are to compete to trigger the schedule. The property value should be an integer and the default value is 2.
- **Cluster Scheduler Lease Valid Time**
  Specifies the period of time that the clustered server can hold a lease before releasing it. The property value should be an integer in seconds and the default value is 300.
- **Cluster Scheduler Lease Check Interval**
  Specifies the time interval the cluster will use to check the number of active leases. The property value should be an integer in seconds and the default value is 30.

Cluster Storage History Number of Copies

Specifies how many copies will be made in the cluster when a new file or folder is added to the history folder. The property value should be an integer and the default value is 2.

Cluster Storage Realm Number of Copies

Specifies how many copies will be made in the cluster when a new file or folder is added to the realm folder. The property value should be an integer and the default value is 2.

Cluster Storage CRD Result Number of Copies

Specifies how many copies will be made in the cluster when a new file or folder is added to the cached report data (CRD) result folder. The property value should be an integer and the default value is 2.

Cluster Memory Storage Number of Copies

Specifies how many memory copies will be shared in the cluster. The property value should be a positive integer and the default value is 2.

If there are quite a few clustered servers in a JReport cluster, sharing memory in all cluster nodes would increase the network load exceedingly which may lead to poor server performance and scalability.

Notify via E-mail When a Server Is Down

Specifies whether to notify somebody via e-mail when a server in the cluster is down.

- **E-mail Address**
  Specifies the e-mail addresses of the people to whom you want to send a notification e-mail when a server in the cluster is down.

Properties Directory

Specifies the properties directory for storing system data. The properties directory of each clustered server should better point to a different physical location.

Realm Directory

Specifies the realm directory for storing system data. The realm directory of each clustered server should better point to a different physical location.

Resource Root

Specifies the report root directory for storing system data. The resource root of each clustered server should better point to a different physical location.

History Directory

Specifies the directory for storing report/catalog/result files. The history directory of each clustered server should better point to a different physical location.

Temporary Files Directory

Specifies the directory for storing temporary result files. Along with the running of JReport Server, temporary result files (generated by background run) will be written to the temp directory. Each clustered server should use an individual temp folder to store its temporary files generated during its working process.

Server's RMI Host

Specifies the RMI IP address or host name of the clustered server.

Server's RMI Port

Specifies the RMI port number of the clustered server.

Enable Cluster

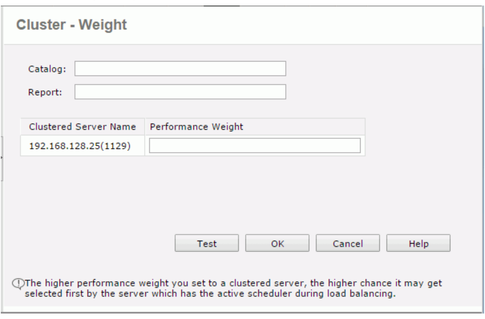
Enables cluster on the server.

Save

Applies any changes to the server.

Weight panel

Use this tab to configure the performance weight on the clustered server.



Catalog

Specifies the catalog that will be used for performance weight testing.

Report

Specifies the report that will be used for performance weight testing.

Clustered Server Name

Displays the name of the clustered server.

Performance Weight

If you have chosen the Weighted Min-load (loadbalance.type=2) algorithm for load balancing, you will have to configure a performance weight for each clustered server. See Configuring performance weight for how to set performance weight and how this algorithm works.

Performance weight is a positive float number. The higher performance weight you set to a clustered server, the higher chance it may get selected by the server which has the active scheduler during load balancing.

Test

Calculates each clustered server's performance weight value at current time.

OK

Applies the settings.

Cancel

Cancels the settings.

Help

Displays the help document about this feature.

Related topics:

- Setting up and starting a JReport Server Cluster
- Managing JReport Server Clusters

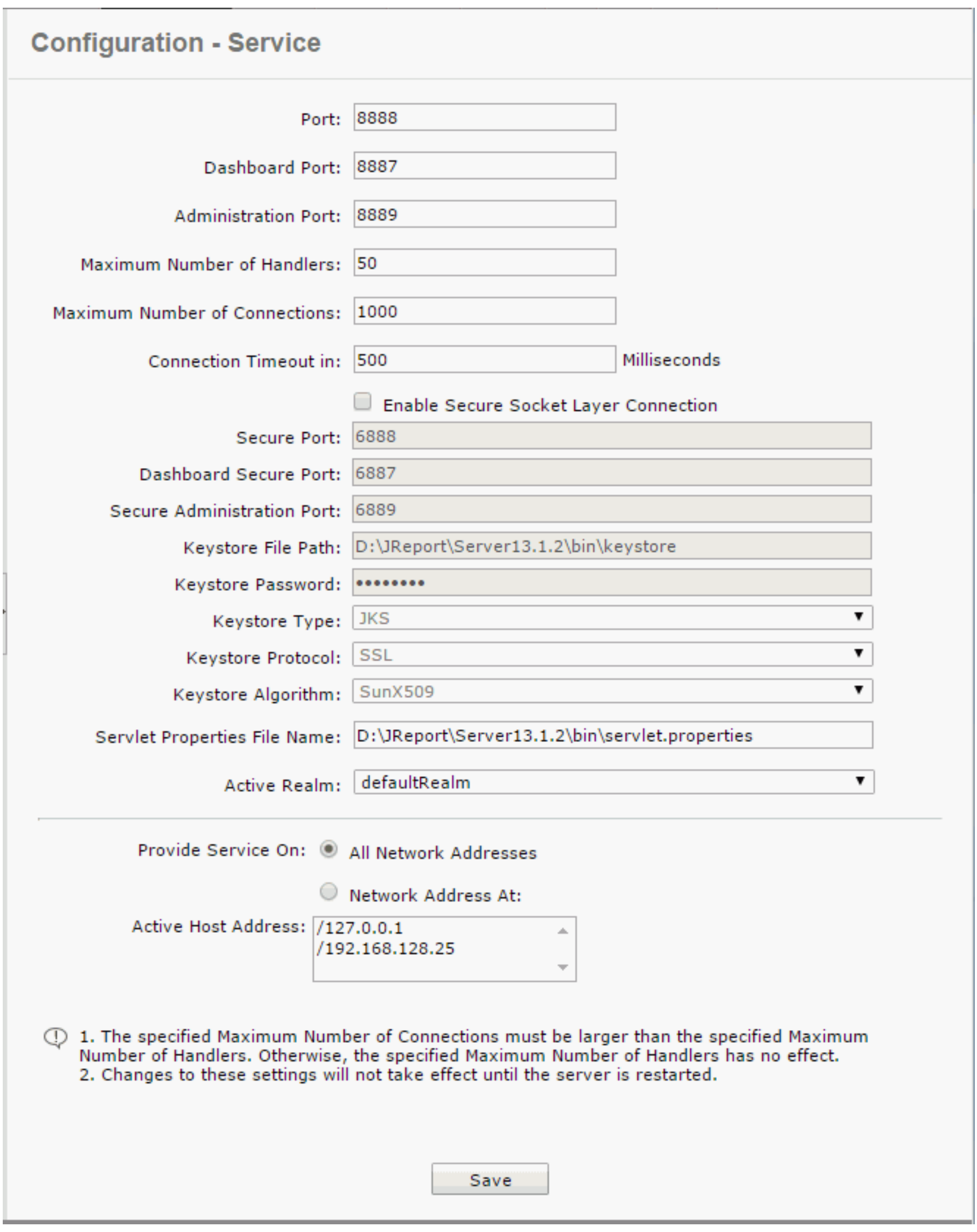
Configuration dialog

After you click Configuration on the system toolbar of the JReport Administration page, the entries to the following panels are available on the drop-down menu. The panels help you to configure JReport Server to suit your own requirements.

- Service panel
- Log panel
- Performance panel
- Advanced panel
- LDAP panel
- Export panel
- Connection panel
- Connection Pool panel
- Upload panel
- Custom Field panel
- Dashboard Listener panel
- Dynamic Connections panel
- Dynamic Display Names panel
- Organizations panel
- E-mail Server panel

Service panel

This panel allows you to configure the service settings of JReport Server.



The following are the options available in the Service panel:

Port

Specifies a port as the TCP port on which JReport Server listens.

When you start JReport Server, you will need to specify a port as the TCP port on which the server listens. The port can be an integer between 1 and 65535. However, numbers less than 1024 are usually reserved by system, and so numbers larger than 1024 are preferred. JReport Server uses 8888 as the default port.

Dashboard Port

Specifies the port for accessing JDashboard page.

Administration Port

Specifies the port used for remote administration. If JReport Server is running as a standalone web server, the administrator can specify the port for client users to configure and administer the server remotely. This port is where the servlet for remote administration (jet.server.servlets.RcServlet) listens. The default administration port is 8889.

Maximum Number of Handlers

Specifies the maximum number of connection handlers.

A connection is set up between the server-side and a client-side when a request from the client-side reaches the server. The maximum number of requests that can be handled depends on the maximum number of handlers. When there is a limit on the maximum number of connection handlers, if there are no free connection handlers available, a request from the client-side will be blocked. In which case, the request will either be handled after a connection handler has been set free, or be refused when timeout occurs. A connection handler is set free after the server sends a response to a client request.

Maximum Number of Connections

Specifies the maximum number of HTTP connections between the server-side and client-side.

The maximum number of connections depends on the number of requests that can be handled. It should be larger than the maximum number of handlers. For example, if the maximum number of handlers is 10, and the maximum number of connections is 12, when the eleventh and twelfth requests come, they will be blocked until a handler has been set free. When the thirteenth request comes, it will be refused.

Connection Timeout in _ Milliseconds

Specifies the maximum time in milliseconds for a request from a client-side to be blocked before being refused by the server. A request will be blocked if there are no free connection handlers in the server. However, it cannot be blocked forever, and if there are still no free connection handlers after the time specified here (in milliseconds), then the request will be refused back to the client.

Enable Secure Socket Layer Connection

If enabled, you can use HTTPS schema to visit JReport Server UI in the standalone mode.

- **Secure Port**
  The port used for visiting JReport Console page via HTTPS schema. It should be different from the Port for HTTP schema.
- **Dashboard Secure Port**
  The port used for visiting JDashboard page via HTTPS schema. It should be different from the Dashboard Port for HTTP schema.
- **Secure Administration Port**
  The port used for visiting JReport Administration page via HTTPS schema. It should be different from the Administration Port for HTTP schema.
- **Keystore File Path**
  The location of your trusted keystore file.
- **Keystore Password**
  The password used to protect the integrity of the keystore.
- **Keystore Type**
  The type of keystore to be instantiated. The valid values are JKS and PKCS12.
- **Keystore Protocol**
  The encryption/decryption protocol to be used on the socket. The valid values are SSL and TLS.
- **Keystore Algorithm**
  The X509 algorithm to use. This defaults to the Sun implementation (SunX509). For IBM JVMs you should use IbmX509.

Servlet Properties File Name

Displays the full path of the property file of the servlet jet.server.servlet.JRServlet.

Assuming that JReport Server has been installed to `C:\JReport\Server`, the servlet property file is `C:\JReport\Server\bin\servlet.properties`.

Active Realm

Specifies the realm that will take effect when the server starts up.

A realm is the context of JReport Server where the entities reside. A user is a kind of entity that is contained in a realm. Different users in one realm must have different names. There can be multiple realms in the server, but only one is active at runtime. Only the users and resources in the active realm are accessible. Realm names cannot contain the "/" or "\" character.

Provide Service On

Specifies whether JReport Server listens on all network addresses or just some.

- **All Network Addresses**
  If checked, JReport Server listens on all network addresses, which means that all the hosts of the machine are active, and the client-end can connect with any of the hosts of this server.
- **Network Address At**
  If checked, JReport Server listens on the specified hosts, and you can specify them by inputting the host names or IP addresses.

Notes:

- The machine that JReport Server runs on can be multi-homed (for example, two interface cards have been installed on the machine), if there is more than one IP address. JReport Server opens the listening port at host name 'localhost' or at IP address '127.0.0.1' automatically.
- When Network Address At is checked, if you want to specify all the host addresses to be active, input * into this field; if you want to specify more than one address to be active, you should separate them using a blank, for example, "leo 204.177.148.110".

Active Host Address

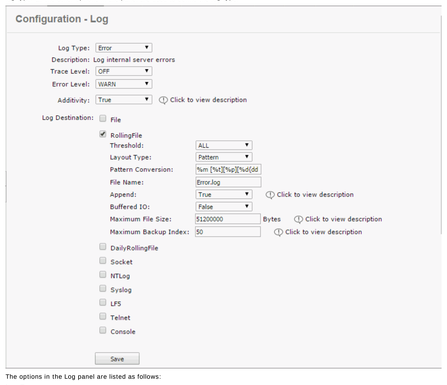
Lists the current active host' address.

Save

Applies changes to JReport Server.

Log panel

Logs can help you find out the reasons of problems, track down the workflow of a business process, and debug trivial messages. Options for configuring JReport Server logs have been provided in this panel. You can configure settings for each log category. After you select a log type, all the other options in the panel are used to define the log type.



The options in the Log panel are listed as follows:

Log Type

Specifies the log category. It can be specified as one of the following:

- **Engine**
  Uses the "Engine" log if you want to log the events related to running reports, that is creating report results or exporting reports. Reports are run by the Engine component of JReport Server.
- **Page Report**
  Uses the "Page Report" log if you want to log the events related to end users modifying and saving page reports, including Ad Hoc and analysis features supported by Page Report Studio.
- **Access**
  Uses the "Access" log if you want to log which users accessed which services, such as report running service and task scheduling service.
- **Manage**
  Uses the "Manage" log if you want to log the events related to modifying settings in the Administration Console or server.properties.
- **Error**
  Uses the "Error" log if you want to log errors in any of the categories.
- **Event**
  Uses the "Event" log if you want to log the events related to the lifecycle of the server instance, such as its start time and stop time.
- **Debug**
  Uses the "Debug" log if you want to log the events most likely needed for debugging purposes, such as the exact SQL statements used to query the database.
- **Performance**
  Uses the "Performance" log if you want to analyze the performance of report result or export operations.
- **Dump**
  Uses the "Dump" log if you want to log events related to at what time what action is started or ended during the process of running tasks. For example, when the task is submitted, when the task is run, when the Engine is initiated, and when the Engine is stopped.

Descriptions

Displays the brief description of the specified log type.

Trace Level

Used for logging something expectable or regular, such as tracing program workflow, logging runtime information and associated elements.

You can set the trace level of the specified log type to one of the following:

- **OFF**
  Specifies not to use the trace level.
- **OUTLINE**
  The OUTLINE level specifies an outline of program workflow, and dump global variables, including a single thread, multiple threads, the time for when to begin to fetch data, and success in exporting to the specified result format.
- **INFO**
  The INFO level specifies informational messages that highlight the application progress at a coarse-grained level, and important local variables, such as query, parameter, formula value used when running reports, connection information, and SQL statement.
- **TRIVIAL**
  The TRIVIAL Level specifies fine-grained informational events most useful in tracing an application. Such as the report structure dump, or the result set dump.

Error Level

Used for logging something unexpected or irregular. For example, a URL is unreachable, a file is not there, or a table cannot be found in a data source.

You can set the error level of the specified log type to one of the following:

- **OFF**
  Specifies not to use the error level.
- **FATAL**
  The FATAL level specifies severe error events that will presumably lead the application to abort, for example, failing to read the valid key, and exceptions that result in feature uncompleted.
- **ERROR**
  The ERROR level specifies error events that may allow the application to continue running, such as failing to load a report, failing to find a catalog file, failing to parse a parameter, and failing to create db buffer.
- **WARN**
  The WARN level specifies potentially harmful situations, such as failing to find the resource, or having found invalid query or formula with grammar error when loading a catalog.

Additivity

Determines the inheritance of the appenders defined by the Logger's ancestors.

Log Destination

Specifies where the log contents will be outputted to.

If you have set log destination, the threshold level for each goal appender defines the basic log level which will be further filtered by JReport trace level and error level, and then comes out the final log output. Trace level and error level are JReport specific log levels developed based on Log4J. They group log information into two categories and add more levels to classify log information.

For example, if threshold=OFF, the trace level and error level will be ignored, and the log information output is none. If threshold=ALL, the output is determined by the trace level and error level. If threshold=one of the other levels, the output will be the result based on the trace level and error level that filter the threshold level.

- **File**
  Outputs log contents to the file appender that requires a layout. Logs of the specified category are written into a file. When File is checked, the following options are available:
  - **Threshold**
    Defines the appender's level (Log4j named as threshold).
  - **Layout Type**
    Specifies the layout type used to format the log contents.
  - **Pattern Conversion**
    Sets the conversion pattern for the Pattern Layout.
  - **File Name**
    Specifies the name of the log file to which the File Appender will output the log contents. The suffix of the log file name is .log.
  - **Append**
    Specifies whether or not to retain the old contents of the specified file.
  - **Buffered IO**
    Specifies to create a buffer for the log IO.
- **Rolling File**
  Outputs log contents to the rolling file appender that requires a layout. When Rolling File is checked, the following options are available:
  - **Threshold**
    Defines the appender's level.
  - **Layout Type**
    Describes the layout type used to format the log contents.
  - **Pattern Conversion**
    Sets the conversion pattern for the Pattern Layout.
  - **File Name**
    Specifies the name of the log file to which the appender will output the log contents. The suffix of the log file name is .log.
  - **Append**
    Specifies whether or not to retain the old contents of the specified file.
  - **Buffered IO**
    Specifies to create a buffer for the log IO.
  - **Maximum File Size**
    Decides the maximum file size of the rolling file appender. The log system will create a new rolling file when the file size exceeds the maximum file size.
  - **Maximum Backup Index**
    Decides the maximum number of the latest rolling files that will be retained by the log system.
- **Daily Rolling File**
  Outputs log contents to the daily rolling file appender that requires a layout. When Daily Rolling File is checked, the following options are available:
  - **Threshold**
    Defines the appender's level.
  - **Layout Type**
    Specifies the layout type used to format the log contents.
  - **Pattern Conversion**
    Sets the conversion pattern for the Pattern Layout.
  - **File Name**
    Specifies the name of the log file to which the appender will output the log contents. The suffix of the log file name is .log.
  - **Append**
    Specifies whether or not to retain the old contents of the specified file.
  - **Buffered IO**
    Specifies to create a buffer for the log IO.
  - **Date Pattern**
    Specifies the data pattern used to generate the daily rolling file.
- **Socket**
  Outputs log contents to a remote log server. The socket appender needs not a layout. When Socket is checked, the following options are available:
  - **Threshold**
    Defines the appender's level.
  - **Remote Host**
    Specifies the host name where the Socket Server is located.
  - **Port**
    Specifies the port number on which the Socket Server listens.
  - **Delay**
    Specifies the timeout interval value for when a client attempts to create a socket connection.
  - **Location Information**
    Specifies whether or not to output the log location information to the socket stream.
- **NTLog**
  Outputs log contents to the NT event log system. Applies only to Windows servers. This appender requires a layout. When NTLog is checked, the following options are available:
  - **Threshold**
    Defines the appender's level.
  - **Layout Type**
    Specifies the layout type used to format the log contents.
  - **Pattern Conversion**
    Sets the conversion pattern for the Pattern Layout.
  - **Source**
    Specifies the source name used and displayed by the NTEvent Viewer.
- **Syslog**
  Outputs log contents to a remote syslog daemon. Applies only to Unix and Linux servers. This appender requires a layout. When Syslog is checked, the following options are available:
  - **Threshold**
    Defines the appender's level.
  - **Layout Type**
    Specifies the layout type used to format the log contents.
  - **Pattern Conversion**
    Sets the conversion pattern for the Pattern Layout.
  - **Syslog Host**
    Specifies the host name where the Syslog server is located.
  - **Facility**
    Specifies the facility name that is used by the Syslog.
  - **Facility Printing**
    Specifies whether or not to print the facility information.
- **LFS**
  Outputs log contents to a swing based logging console. LFS has a swing-based UI to view all of the logging messages at runtime and enables you to search and filter logging messages. This appender needs not a layout.

  When LFS is checked, the following option is available:
  - **Threshold**
    Defines the appender's level.
- **Telnet**
  Outputs log contents to a read-only socket. You use telnet to connect to the socket and receive log data, and this is handy for remote monitoring, especially when monitoring a servlet. This appender requires a layout.

  When Telnet is checked, the following options are available:
  - **Threshold**
    Defines the appender's level.
  - **Layout Type**
    Specifies the layout type used to format the log contents.
  - **Pattern Conversion**
    Sets the conversion pattern for the Pattern Layout.
  - **Port**
    Specifies the port number on which Telnet listens.
- **Console**
  Outputs log contents to the standard stream of Jave console. When Console is checked, the following options are available:
  - **Threshold**
    Defines the appender's level.
  - **Layout Type**
    Specifies the layout type that is used to format the log contents.
  - **Pattern Conversion**
    Sets the conversion pattern for the Pattern Layout.
  - **Target**
    Specifies the standard IO target of the Console.
    - **System.out**
      If selected, the log contents will be outputted to the standard output stream of the console.
    - **System.err**
      If selected, the log contents will be outputted to the standard error stream of the console.

Save

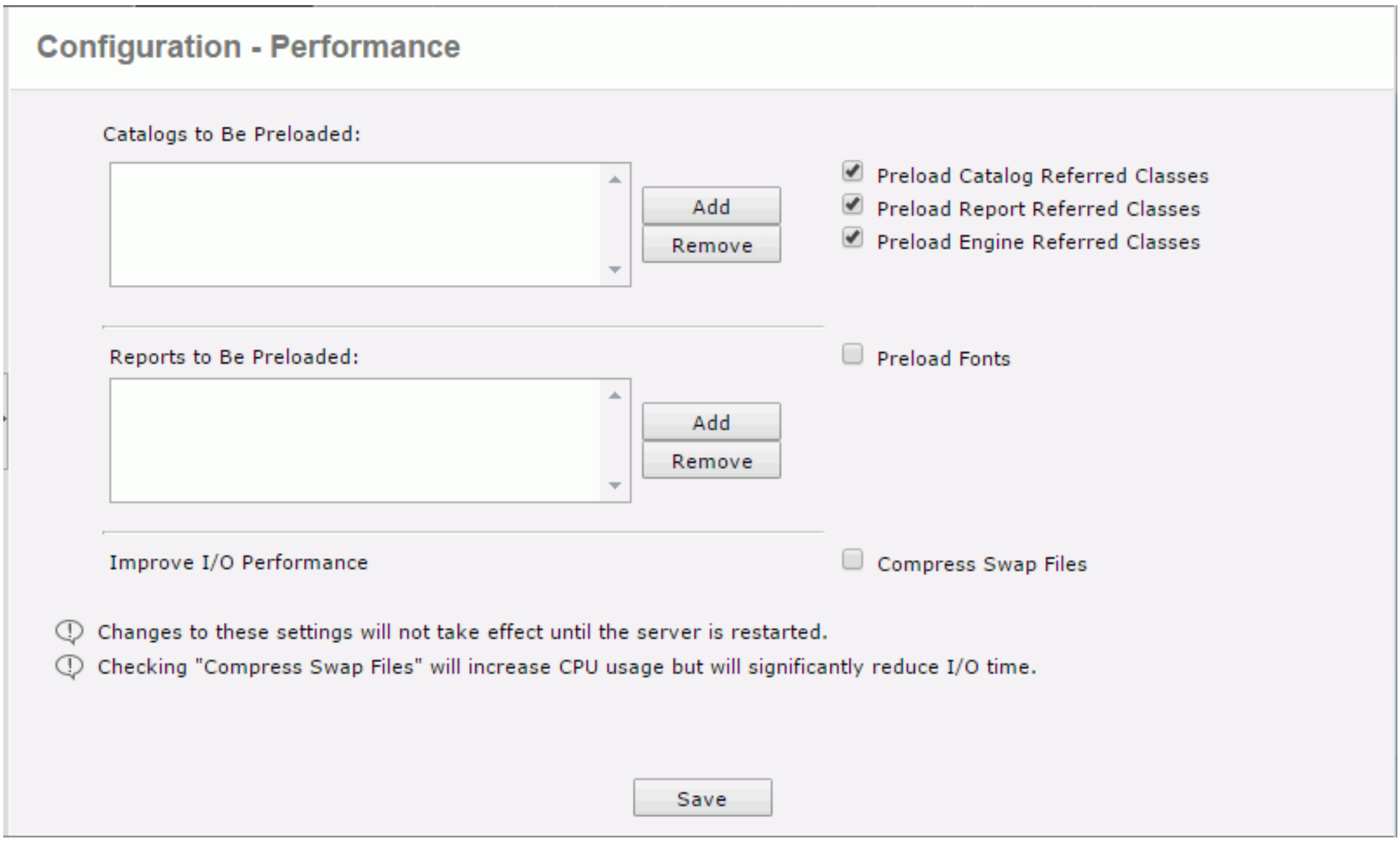
Applies all changes to JReport Server.

Notes:

- The Pattern Conversion option is available only when the Pattern is chosen as the Layout Type.
- For more information about the Pattern Conversion option, see comments in the LogConfig.properties file located in `<install_root>\bin` directory.

Performance panel

This panel allows you to tune the performance of JReport Server.



The options in this panel are as follows:

Catalogs to Be Preloaded

Displays the catalogs that will be loaded when JReport Server is started.

- **Add**
  Adds a catalog to be preloaded.
- **Remove**
  Removes a catalog from the preloading list.
- **Preload Catalog Referred Classes**
  Specifies to preload the Java classes that are used by a catalog when JReport Server is started.

  Preloading some useful Java classes will improve performance. Otherwise, it will take some time for JReport Server to load any required Java classes when generating reports.

  This option is not available to organization admin.
- **Preload Report Referred Classes**
  Specifies to preload the Java classes that are used by a report when JReport Server is started. Not available to organization admin.
- **Preload Engine Referred Classes**
  Specifies to preload the Java classes used by the JReport Engine when generating reports when JReport Server is started. Not available to organization admin.

Reports to Be Preloaded

Displays the reports that will be loaded when JReport Server is started.

- **Add**
  Adds a report to be preloaded.
- **Remove**
  Removes a report from the preloading list.
- **Preload Fonts**
  Specifies to preload fonts when JReport Server is started. Not available to organization admin.

Improve I/O Performance

This option is not available to organization admin.

- **Compress Swap Files**
  Specifies whether or not to compress the temporary data generated during runtime before it is swapped to disk. By compressing the swap files, the I/O efforts in certain circumstance may remarkably reduced so that the overall performance can be improved.

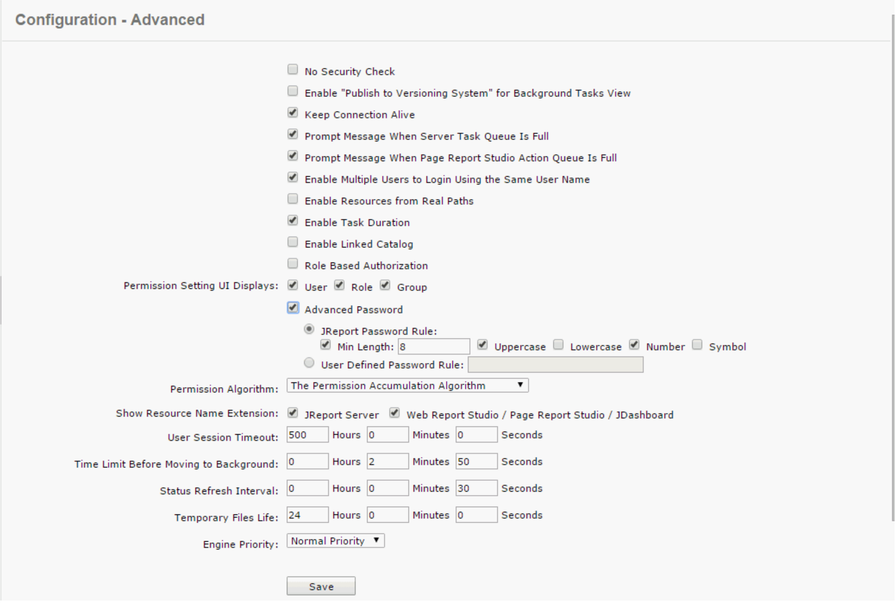
Note: Compressing swap files will increase CPU pressure because it uses compress algorithm to shrink data, so if your system already has high CPU usage, enabling this option will bring extra performance impact, depending on different circumstance, and such impact may overcome the performance gain that comes from reducing I/O time.

Save

Applies the changes to JReport Server.

Advanced panel

This panel allows you to configure the advanced settings of JReport Server.



The options available in this panel are as follows:

No Security Check

Specifies whether the server responds to all the requests without security checking.

Enable "Publish to Versioning System" for Background Tasks View

Specifies whether to generate the hyperlink Publish to Versioning System in the view page when you run a report on the JReport Console page. Click this hyperlink to save the report result as a version.

Keep Connection Alive

Specifies whether to keep the connection alive. By default, this option is checked. When connections are established to the JReport Server at runtime, you can choose either to keep the connections alive or to close them. If you uncheck this option, the connection will be disconnected once the HTTP request has been responded to. That is, Internet Explorer will not hold the open connection.

Prompt Message When Server Task Queue Is Full

If checked, a message will be prompted asking you either to place the task in the waiting queue or to cancel it when a task cannot be run in JReport Server immediately due to the task queue being fully occupied.

Prompt Message When Page Report Studio Action Queue Is Full

If checked, a message will be prompted asking you either to place the action in the waiting queue or to cancel it when an action cannot be performed in Page Report Studio immediately due to the action queue being fully occupied, or simply showing that no more action can be held at the moment when the waiting queue is full.

Enable Multiple Users to Login Using the Same User Name

Specifies whether multiple sessions created by one user can exist simultaneously. By default, this option is checked and one user can log onto JReport Server repeatedly without having to terminate former sessions until the session number reaches the number permitted by the license. If this option is unchecked, only the newly created session will be active, and all the other former sessions created by the same user will be terminated by the server automatically.

Enable Resources from Real Paths

Specifies whether to enable getting resources from real paths.

Enable Task Duration

Specifies whether to enable setting a time duration for a task and asking JReport Server to cancel the task or to notify you or someone else of the task status via e-mail if the task has not yet finished running when the task duration is up.

If this option is checked, the Duration tab will be available in the Advanced Run and Schedule dialogs.

Enable Linked Catalog

Enables to predefine a linked catalog at server level for the My Components, Public Components, My Reports and Public Reports folders to inherit.

If you have set a linked catalog here, these folders can inherit the linked catalog from the server level once their Enable Linked Catalog property is enabled.

If the option is unchecked, these folders cannot inherit linked catalog.

- **Select Another Catalog**
  Specifies another catalog in the Select Another Catalog dialog.

Role Based Authorization

Specifies whether to activate the role based security in replace of the default applied security of setting permissions for users, groups and roles. In the role based security, permissions are defined on roles only, and users and groups are mapped to roles.

If this option is checked, the option Permission Setting UI Displays will be hidden since it applies to the default applied security.

Permission Setting UI Displays

Controls whether to display user, role or group permissions in the permission setting UI for the server resource tree node and version properties, and the scheduling to version and advanced run dialogs.

Advanced Password

Specifies whether to apply advanced password rule on JReport Server. When this option is enabled, when setting password on JReport Server, a complex one which matches the rule specified below is required. However, this rule doesn't affect administrator's changing and setting password for other users.

- **JReport Password Rule**
  Specifies to use JReport's built-in password rule.
  - **Min Length**
    Specifies the minimum number of characters that can be used in the password. The number specified here should be larger than the password minimum length defined for all users created on the Security > User panel. The default number is 8.
  - **Uppercase**
    If checked, the password must contain at least one uppercase letter.
  - **Lowercase**
    If checked, the password must contain at least one lowercase letter.
  - **Number**
    If checked, the password must contain at least one Arabic numeral.
  - **Symbol**
    If checked, the password must contain at least one character that is not defined as letter, numeral or space.
- **User Defined Password Rule**
  JReport also provides you with the interface AdvancedPasswordRule in the package jet.server.api.admin for defining your own password rule. You can specify a Java class to implement this interface, then add the path of the class file to the ADDCLASSPATH variable in setenv.bat, which is located in `<install_root>\bin`. After that, check this option and enter the name of the class in the text box to validate the password.

Permission Algorithm

Specifies an algorithm for calculating a user, role or group's permission.

For information about the permission accumulation algorithm and the user permission replacement algorithm, see server.permission.algorithm.

Show Resource Name Extension

Specifies whether to show the resource name extension in the resource tree.

- **JReport Server**
  Specifies whether to show the resource name extension in the resource tree on the JReport Server Console page.
- **Web Report Studio / Page Report Studio / JDashboard**
  Specifies whether to show the resource name extension in the resource tree on Web Report Studio, Page Report Studio and JDashboard.

User Session Timeout

Specifies the time limit for a user to gain access to JReport Server.

Generally speaking, if several users attempt to gain access to JReport Server at the same time and among them, a user is unable to send requests to JReport Server for a period of time, the server will then log it out after timeout. The default session timeout is 500 hours.

Time Limit Before Moving to Background

Specifies the maximum time for running a report. If a report is so large that JReport Server cannot run it within the time specified here, it will be put into the Background Tasks page and the server will go on running the report there. If you want to stop running the report, select the report by clicking in the row and then delete it from the Background Tasks tab.

Status Refresh Interval

Specifies task duration check frequency. By default, JReport Server checks task duration every 30 seconds. This value can also be reset by the task.duration.check_cycle property in the server.properties file. Note that the value must be an integer greater than 0.

The option is available when the option Enable Task Duration in this panel is checked.

Temporary Files Life

Specifies the length of the period to keep the temp files for after they have been generated. By default it is 24 hours. Temp files are deleted 24 hours after they have been generated.

Engine Priority

Specifies the priority of the engine in CPU usage when multiple tasks are running on your machine. It can be one of the following: Lowest Priority, Low Priority, Normal Priority, High Priority, and Highest Priority.

On the condition that multiple tasks are running on your machine, you can decide your engine's priority in CPU usage by choosing one of them. For example, if you choose Low Priority, JReport Engine will share a comparatively small part of the CPU usage when it is running. Whereas, if you choose High Priority, JReport Engine will occupy more CPU usage. This choice provides you with greater flexibility in adjusting the engine's utilization of the CPU according to your requirement. Of course, the higher the priority, the faster the performance.

Save

Applies the changes to JReport Server.

LDAP panel

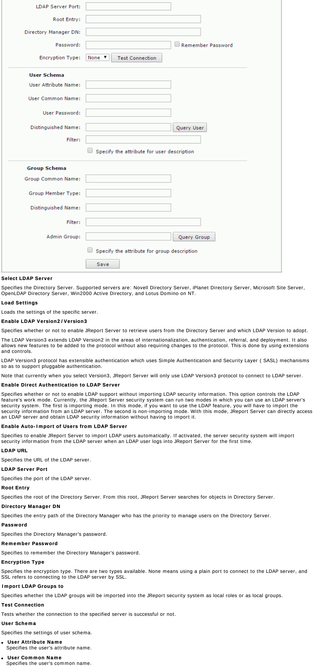
LDAP (Lightweight Directory Access Protocol) is a lightweight client-server protocol for accessing directory servers. With LDAP support, the JReport Server enables you to import users/groups from directory servers.

This panel allows you to configure the JReport Server LDAP settings. It contains four tabs:

- Server
- Import
- Synchronize
- Role Map

Server

This tab allows you to configure LDAP server settings.



Select LDAP Server

Specifies the Directory Server. Supported servers are: Novell Directory Server, iPlanet Directory Server, Microsoft Site Server, OpenLDAP Directory Server, Windows Active Directory, and Lotus Domino on NT.

Load Settings

Loads the settings of the specific server.

Enable LDAP Version2/Version3

Specifies whether or not to enable JReport Server to retrieve users from the Directory Server and which LDAP Version to adopt.

The LDAP Version3 extends LDAP Version2 in the areas of internationalization, authentication, referral, and deployment. It also allows new features to be added to the protocol without also requiring changes to the protocol. This is done by using extensions and controls.

LDAP Version3 protocol has extensible authentication which uses Simple Authentication and Security Layer (SASL) mechanisms so as to support pluggable authentication.

Note that currently when you select Version3, JReport Server will only use LDAP Version3 protocol to connect to LDAP server.

Enable Direct Authentication to LDAP Server

Specifies whether or not to enable LDAP support without importing LDAP security information. This option controls the LDAP feature's work mode. Currently, the JReport Server security system can run two modes in which you can use an LDAP server's security system. The first is importing mode. In this mode, if you want to use the LDAP feature, you will have to import the security information from an LDAP server. The second is non-importing mode. With this mode, JReport Server can directly access an LDAP server and obtain LDAP security information without having to import it.

Enable Auto-Import of Users from LDAP Server

Specifies to enable JReport Server to import LDAP users automatically. If activated, the server security system will import security information from the LDAP server when an LDAP user logs into JReport Server for the first time.

LDAP URL

Specifies the URL of the LDAP server.

LDAP Server Port

Specifies the port of the LDAP server.

Root Entry

Specifies the root of the Directory Server. From this root, JReport Server searches for objects in Directory Server.

Directory Manager DN

Specifies the entry path of the Directory Manager who has the priority to manage users on the Directory Server.

Password

Specifies the Directory Manager's password.

Remember Password

Specifies to remember the Directory Manager's password.

Encryption Type

Specifies the encryption type. There are two types available. None means using a plain port to connect to the LDAP server, and SSL refers to connecting to the LDAP server by SSL.

Import LDAP Groups to

Specifies whether the LDAP groups will be imported into the JReport security system as local roles or as local groups.

Test Connection

Tests whether the connection to the specified server is successful or not.

User Schema

Specifies the settings of user schema.

- **User Attribute Name**
  Specifies the user's attribute name.
- **User Common Name**
  Specifies the user's common name.
- **User Password**
  Specifies the user's password.
- **Distinguished Name**
  Specifies the name of the organization unit inside the LDAP server where you want to perform a search for users.
- **Query User**
  Views properties of users in the organization unit.
- **Filter**
  Specifies the filter criteria with which to search for users.
- **Specify the attribute for user description**
  Specifies the specific attribute which will be used as JReport user information.
  - **Attribute Name**
    Specifies the attribute name.

Group Schema

Specifies the group schema settings.

- **Group Common Name**
  Specifies a common name for the group.
- **Group Member Type**
  Specifies the member type of the group.
- **Distinguished Name**
  Specifies the name of the organization unit inside the LDAP server where you want to perform a search for groups.
- **Filter**
  Specifies the filter criteria with which to search for groups.
- **Admin Group**
  JReport Server will add the group specified here as a member to the Admin group.
- **Query Group**
  Views properties of groups in the organization unit.
- **Specify the attribute for group description**
  Specifies the specific attribute which will be used as JReport group information.
  - **Attribute Name**
    Specifies the attribute name.

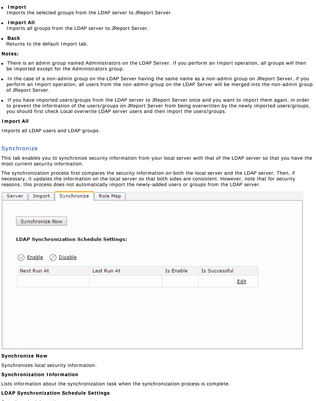
Save

Saves all changes.

There are several examples of LDAP server configuration for your reference.

Import

This tab allows you to import users/groups from directory servers.



LDAP users overwrite local users

Specifies that users of the LDAP server will overwrite those of the local server.

Local users overwrite LDAP users

Specifies that users of the local server will overwrite those of the LDAP server.

List Users

Lists LDAP users that exist in both LDAP server and JReport Server.

List Groups

Lists LDAP groups that exist in both LDAP server and JReport Server.

Import Users

Imports LDAP users.

If LDAP Server overwrite Local users is selected, all LDAP users will then be imported, and any JReport Server users that have the same names as the LDAP users will be overwritten.

If Local overwrite LDAP server users is selected, all LDAP users will be imported, and any users that have the same names as JReport Server users will be overwritten.

- **Import**
  Imports the selected users from the LDAP server to JReport Server.
- **Import All**
  Imports all users from the LDAP server to JReport Server.
- **Back**
  Returns to the default Import tab.

Import Groups

Imports LDAP groups.

If LDAP Server overwrite Local users is selected, all LDAP groups will be imported, any LDAP group that has the same name as a group in the local server will be merged into the local group, and local users of the same names as the LDAP users will be overwritten.

If Local overwrite LDAP server users is selected, all LDAP groups will be imported, any LDAP group that has the same name as a group in the local server will be merged into the local group, and LDAP users of the same names as the local users will be overwritten.

- **Import**
  Imports the selected groups from the LDAP server to JReport Server.
- **Import All**
  Imports all groups from the LDAP server to JReport Server.
- **Back**
  Returns to the default Import tab.

Notes:

- There is an admin group named Administrators on the LDAP Server. If you perform an Import operation, all groups will then be imported except for the Administrators group.
- In the case of a non-admin group on the LDAP Server having the same name as a non-admin group on JReport Server, if you perform an Import operation, all users from the non-admin group on the LDAP Server will be merged into the non-admin group of JReport Server.
- If you have imported users/groups from the LDAP server to JReport Server once and you want to import them again, in order to prevent the information of the users/groups on JReport Server from being overwritten by the newly imported users/groups, you should first check Local overwrite LDAP server users and then import the users/groups.

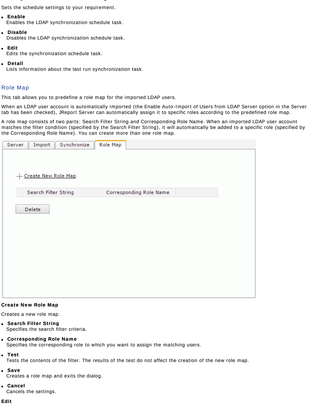
Import All

Imports all LDAP users and LDAP groups.

Synchronize

This tab enables you to synchronize security information from your local server with that of the LDAP server so that you have the most current security information.

The synchronization process first compares the security information on both the local server and the LDAP server. Then, if necessary, it updates the information on the local server so that both sides are consistent. However, note that for security reasons, this process does not automatically import the newly-added users or groups from the LDAP server.



Synchronize Now

Synchronizes local security information.

Synchronization Information

Lists information about the synchronization task when the synchronization process is complete.

LDAP Synchronization Schedule Settings

Sets the schedule settings to your requirement.

- **Enable**
  Enables the LDAP synchronization schedule task.
- **Disable**
  Disables the LDAP synchronization schedule task.
- **Edit**
  Edits the synchronization schedule task.
- **Detail**
  Lists information about the last run synchronization task.

Role Map

This tab allows you to predefine a role map for the imported LDAP users.

When an LDAP user account is automatically imported (the Enable Auto-Import of Users from LDAP Server option in the Server tab has been checked), JReport Server can automatically assign it to specific roles according to the predefined role map.

A role map consists of two parts: Search Filter String and Corresponding Role Name. When an imported LDAP user account matches the filter condition (specified by the Search Filter String), it will automatically be added to a specific role (specified by the Corresponding Role Name). You can create more than one role map.

Create New Role Map

Creates a new role map.

- **Search Filter String**
  Specifies the search filter criteria.
- **Corresponding Role Name**
  Specifies the corresponding role to which you want to assign the matching users.
- **Test**
  Tests the contents of the filter. The results of the test do not affect the creation of the new role map.
- **Save**
  Creates a role map and exits the dialog.
- **Cancel**
  Cancels the settings.

Edit

Edits the specified host map.

Test

Tests the contents of the filter.

Delete

Deletes the specified role map.

Related topics:

- Using an LDAP server's security system

Export panel

This panel allows you to configure the default settings for exporting report result in JReport Server. It is divided into three tabs:

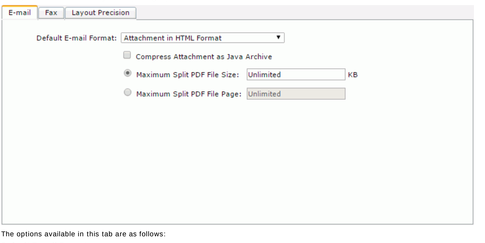
- E-mail tab
- Fax tab
- Layout Precision tab

Save

Saves changes made in this panel. The changes will be applied the next time JReport Server starts.

E-mail tab

Configures the default settings for exporting report result to e-mail.



The options available in this tab are as follows:

Default E-mail Format

Specifies how a published report is sent out by e-mail. The format can be any of the following:

- **E-mail Result in HTML E-mail Format**
  Sends the report result via e-mail to the specified address in HTML format. The report result will be shown in HTML format in the mail body.
- **E-mail Result in Plain Text E-mail Format**
  Sends the report result via e-mail to the specified address in plain text format. The report result will be shown in plain text format in the mail body with no other information such as color, images and so on.
- **Attachment in JReport Result Format**
  Sends the report result via e-mail to the specified address with a JReport result file as attachment.
- **Attachment in HTML Format**
  Sends the report result via e-mail to the specified address with an HTML file as attachment.
- **Attachment in PDF Format**
  Sends the report result via e-mail to the specified address with a PDF file as attachment.
- **Attachment in Excel Format**
  Sends the report result via e-mail to the specified address with an Excel file as attachment.
- **Attachment in Text Format**
  Sends the report result via e-mail to the specified address with a Text file as attachment.
- **Attachment in RTF Format**
  Sends the report result via e-mail to the specified address with a RTF file as attachment.
- **Attachment in XML Format**
  Sends the report result via e-mail to the specified address with an XML file as attachment.
- **Attachment in PostScript Format**
  Sends the report result via e-mail to the specified address with a PostScript file as attachment.

Compress Attachment as Java Archive

Specifies whether or not to compress the mail attachment as Java Archive.

Maximum Split PDF File Size (KB)

Specifies in KB the largest size each PDF file could have after splitting.

Maximum Split PDF File Page

Specifies how many pages each PDF file could have at most after splitting.

If you choose to split a PDF file by file size, then how to split the PDF file depends on the following two aspects:

- The separated pages by the before-split PDF file
- The maximum file size specified for an after-split PDF file

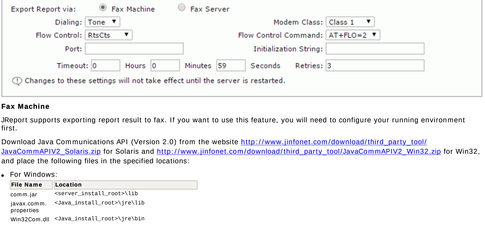
Here the before-split PDF file refers to the big PDF file to be split and an after-split PDF refers to one of the smaller PDF files generated after splitting the big PDF file. When a PDF file is to be split by file size, the splitting will be carried out based on the pages the file separates but not physically on the maximum file size specified for an after-split PDF file. However, the maximum file size helps to decide by which page to split: the page that the maximum size comes to is not included with the previous pages in an after-split file, but instead is the beginning page of the following after-split file. For example, there is a 2M PDF file with 1M per page. If the maximum file size is set to 1.5M, we will get two PDF files with each 1M and one page as the split result.

Notes:

- When the before-split PDF file contains only one page, the split function will not take effect for the only one page cannot be further split either by file size or by file page, and therefore the result is one PDF file as attachment no matter the Split PDF feature is enabled or not.
- The Split PDF function takes effect only when the selected option value is greater than 0.
- If you check to use the Split PDF function, the TOC and Sign options are disabled when scheduling a report result to e-mail in PDF file attachment.

Fax tab

Configures the default settings for exporting report result to fax. You can specify to export the report result either via a fax machine or a fax server.



Fax Machine

JReport supports exporting report result to fax. If you want to use this feature, you will need to configure your running environment first.

Download Java Communications API (Version 2.0) from the website http://www.jinfonet.com/download/third_party_tool/JavaCommAPIV2_Solaris.zip for Solaris and http://www.jinfonet.com/download/third_party_tool/JavaCommAPIV2_Win32.zip for Win32, and place the following files in the specified locations:

- For Windows:

File Name	Location
comm.jar	`<server_install_root>\lib`
javax.comm.properties	`<Java_install_root>\jre\lib`
Win32Com.dll	`<Java_install_root>\jre\bin`

- For Solaris:

File Name	Location
comm.jar	`<server_install_root>/lib`
javax.comm.properties	`<Java_install_root>/jre/lib`
libSolarisSerialParallel.so	`LD_LIBRARY_PATH`

The following are options available when Fax Machine is checked:

- **Dialing**
  Specifies the dialing mode for the fax modem. Options are: Pulse or Tone.
- **Modem Class**
  Specifies the class of the modem. Options are: Class 1, Class 2 or Class 2.0. All are fax protocols. Most modems only support Class 1, so if you select Class 2 or Class 2.0, you should make sure that your modem can support it.
- **Flow Control**
  Specifies the flow control mode between DTE (Data Terminal Equipment) and DCE (Data Circuit-terminating Equipment). Specifying flow control can help the compressing data function of the modem work better.

  The flow control options are as follows:
  - **RtsCts**
    Flow control of the hardware (recommended).
  - **Xon/Xoff**
    Flow control of the software.
  - **None**
    No flow control specified.
- **Flow Control Command**
  Specifies the flow control command according to the modem being used. If not contained in the drop-down list, you can leave this field empty and then enter the command as part of the initial string. The command should be obtained from your modem manual.
- **Port**
  Specifies the port number. The port should be obtained from your modem manual.
- **Initialization String**
  Initializes the modem. The string should be obtained from your modem manual.
- **Timeout**
  Specifies the maximum amount of time that the fax should wait for a response from the destination before timing out.
- **Retries**
  When the line is busy, the report result cannot be faxed. Here, you can specify the maximum number of times the modem re-tries faxing the report result.

Fax Server

If you choose to export the report result via a fax server, configure the following settings according to your requirements:

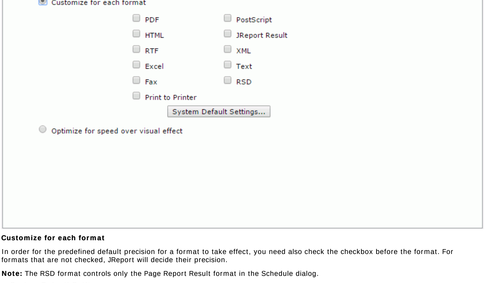
- **Fax Gateway Connector**
  Specifies the name of the class implemented by users.

  Note: By default, the fax server JReport uses is based on Hylafax Server, however, if you want to export your report result via Hylafax Server, you need to download the gnu-hylafax packages according to your requirements from http://sourceforge.net/projects/gnu-hylafax/, for example, gnu-hylafax-util-0.0.9.2.jar, gnu-inet-ftp-0.0.9.2.jar, and gnu-hylafax-0.0.9.2.jar, and then add them to the class path of setenv.bat in the ADDCLASSPATH variable.
- **Server IP**
  Specifies the IP address or domain name of the fax server.
- **Server Port**
  Specifies the port number of the fax server.
- **Login ID**
  Specifies the username for the class communicating with fax server.
- **Password**
  Specifies the password for the class communicating with fax server.
- **Fax Sender**
  Specifies the user's name that shows in the fax server manager.
- **Special Parameters**
  Specifies some parameters for the fax server.
- **Time Out**
  Specifies the maximum amount of time that the fax should wait for a response from the destination before timing out.

  Note: For Hylafax Server the value should not be larger than 59 seconds. It is a limitation of Hylafax Server.
- **Retries**
  Specifies the number of times the modem retries faxing the report result.

Layout Precision tab

Configures the precision level for different formats with which the report result will be exported. The settings of high/low precision take effect only in the report tabs whose Precision Sensitive property is set to true in the Report Inspector of JReport Designer.



Customize for each format

In order for the predefined default precision for a format to take effect, you need also check the checkbox before the format. For formats that are not checked, JReport will decide their precision.

Note: The RSD format controls only the Page Report Result format in the Schedule dialog.

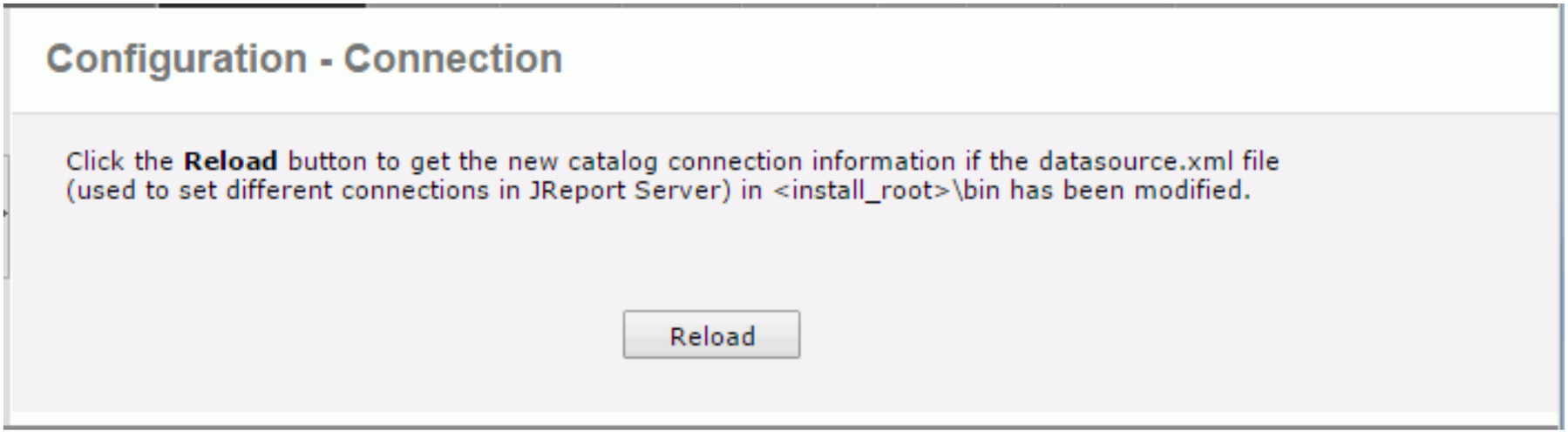
- **System Default Settings**
  Predefines the default precision for each format.
  - **High**
    If checked, the report result will be exported to the specified format with high precision.
  - **Low**
    If checked, the report result will be exported to the specified format with low precision.

Optimize for speed over visual effect

If checked, the report result will be exported with precision level decided by JReport.

Connection panel

This panel is used for retrieving the modified catalog connection information from the datasource.xml file in the `<install_root>\bin` directory.



The option available in this panel is as follows:

Reload

Reloads catalog connection information from the datasource.xml file.

Connection Pool panel

It is used for viewing connection information and for deleting unused connections.

The options available in this panel are as follows:

Disconnect

Cancels the selected connection from working.

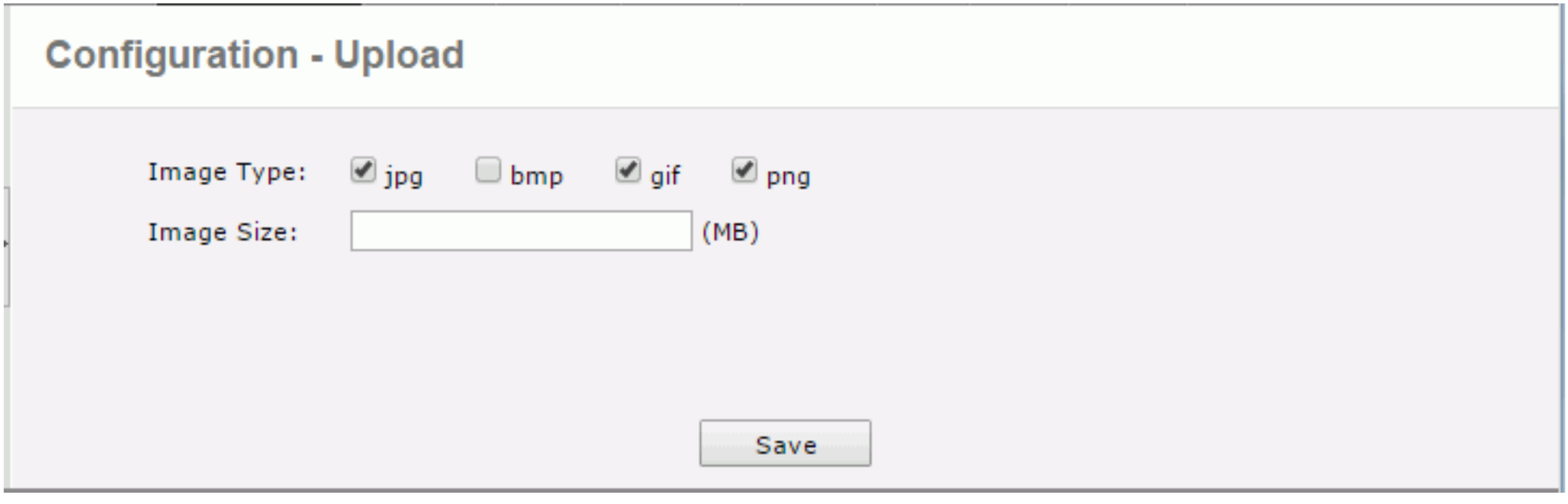
Connection Information Table

The table consists of these columns:

- **User**
  The user who triggered the connection.
- **URL - URL1**
  JDBC URL that can establish a connection to your database. The URL contents are case sensitive.
- **Expiration Time**
  The period of time in seconds it takes for an active connection to expire. "0" means that the active connection will not expire.
- **Idle Expiration Time**
  The period of time for when a free connection can be idle before it is closed. The connection will not be released until the defined time has reached its limit. "1" means that the connection will be released after 1 second. "0" means the connection will be closed right after it starts idling.
- **Maximum Connection Count**
  The maximum number of connections based on the URL. Once the number of the connections reaches the maximum, a new connection will be blocked until a free connection becomes available. Generally speaking, the fewer the connections, the better the performance of JReport Server. "0" means that there will be no limit on the number of the connections.
- **Maximum Share Count**
  The maximum number of requests in one connection that can be executed simultaneously. Once the number of the requests reaches the maximum, a new connection will be created to connect to the database. "1" means that one request in one connection can be executed simultaneously. "0" means that there will be no limit to the maximum number of requests sharing one connection and that all the requests in one connection can be executed simultaneously. The smaller the number of requests, the better the performance of JReport Server.
- **Attempts**
  The number of times the engine will try to establish a new connection. "1" means that the engine will only try once to create a new connection. This property has a close relation to the property Maximum Connection Count. Once the number of the connections reaches the maximum, a new connection will be blocked until a free connection becomes available.
- **Interval**
  The interval the engine will wait for between attempts to create a new connection if a previous try has failed. "0" means that the engine will start the next attempt immediately. The unit of the property value is milliseconds.
- **Last User Connection Time**
  The time when the user connection was last established.
- **Current Idle Time**
  The time for when the free connection has been idle.
- **Current Share Count**
  The number of requests in the connections that are being executed concurrently.

Upload panel

This is used for specifying the allowed image size and types for the images which are uploaded from the local file system when end users insert images into reports in Page Report Studio.



The options available in this panel are as follows:

Image Type

Sets the allowed image types.

Image Size

Sets the allowed image size. If the value is blank, it means that the image size is unlimited.

Save

Saves changes made in this panel. The changes will be applied the next time JReport Server starts.

Custom Field panel

This panel is used for defining and managing user defined fields which can be used as resource properties.

The options available in this panel are as follows:

New Custom Field

Opens the New Custom Field dialog for creating a custom field.

Edit

This option is enabled after you select one or more custom fields in the Custom Field table below. It contains two operations:

- **Edit**
  Displays a dialog to edit the information of the selected custom field.
- **Delete**
  Deletes the selected custom fields.

Custom Field table

Displays the custom field information.

- **Custom Field Name**
  Displays the name of the custom fields.
- **Description**
  Displays the description of the custom fields.
- **Enabled**
  Shows whether the custom fields are enabled or not.

Related topics:

- Working with custom fields

Dashboard Listener panel

This panel is used for managing implementations of the Dashboard Listener API.

The options available in the panel are as follows:

Toolbar

- **New Dashboard Listener**
  Opens the New Dashboard Listener dialog for adding an implementation of the Dashboard Listener API.

- **Properties**
  Modifies the target value of the selected implementation in the displayed dialog. For details about the options in the dialog, refer to Step 2: Specifying target resources.

- **Enable**
  Enables the selected disabled implementations.

- **Disable**
  Disables the selected enabled implementations.

- **Delete**
  Deletes the selected implementations.

Classes table

Lists all implementation classes registered. The classes can be ordered by the column titles in alphabetical order.

The table contains the following columns:

- **Class Name**
  The implementation class name. Each is unique and cannot be null.

- **Target**
  Shows the resources that the implementation will take effect on. When a dashboard has been set as the target value of different implementations, the earliest implementation will take effect on the dashboard.

- **Description**
  The description of the implementation.

- **Enabled**
  Shows whether the implementation is enabled.

The context menu after selecting one or more classes in the table:

- **Properties**
  Modifies the target value of the selected implementation in the displayed dialog. For details about the options in the dialog, refer to Step 2: Specifying target value.

- **Enable**
  Enables the selected disabled implementations.

- **Disable**
  Disables the selected enabled implementations.

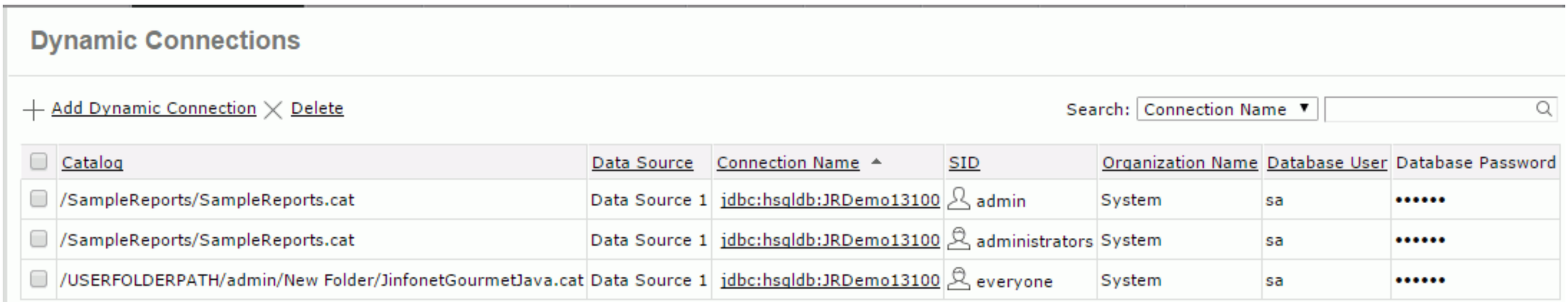
- **Delete**
  Deletes the selected implementations.

Related topics:

- Dashboard Listener API

Dynamic Connections panel

The panel is used for managing dynamic connections.



The options available in this panel are as follows:

Add Dynamic Connection

Creates a dynamic connection in the Add Dynamic Connection dialog.

Delete

Deletes the selected dynamic connections.

Search

Searches for specific dynamic connections.

- **Search category drop-down list**
  Specifies the search category. For example, if you want to find the dynamic connections which use the catalog SampleReprts.cat, select Catalog from the drop-down list.

  There are the following search categories:

  - **Catalog**
    Searches for catalogs.
  - **Connection Name**
    Searches for connection names.
  - **SID**
    Searches for SIDs.
  - **Database User**
    Searches for database users.
- **Keyword text box**

  Specifies the search keyword. After inputting the text, click in the text box to start the search. The **x** button in the text box is used to clear the input text.

Dynamic connection table

All columns are sortable except for Database Password (to sort the table by certain column, click on the column name). By Default the table is sorted by Connection Name.

- **Checkbox**
  Specifies whether or not to select a dynamic connection. Select the checkbox on the column header to select all the dynamic connections.

The table consists of these columns:

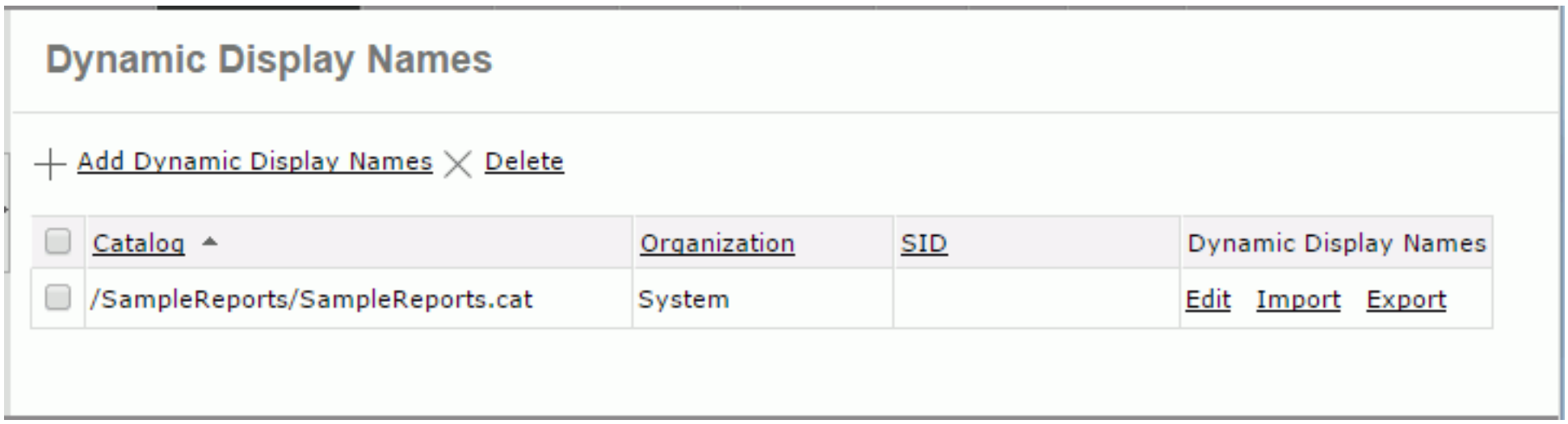
- **Catalog**
  Displays the catalogs of the dynamic connections with the full resource path, for example, `/SampleReports/SampleReports.cat`.
- **Data Source**
  Displays the data source names of the dynamic connections in the catalog.
- **Connection Name**
  Displays the dynamic connection names. Click a connection name to edit the dynamic connection in the Edit Dynamic Connection dialog.
- **SID**
  Displays the security identifiers (SID). An SID can be a group, a role, or a user in the JReport Server security system.
- <**Organization Name**
  Shows the organizations that the SIDs belong to. The column is available to system admin when the organization feature is enabled.
- **Database User**
  Displays the database user names of the dynamic connections. Null means using the default database user name.
- **Database password**
  Displays the database passwords of the dynamic connections. The passwords are masked.

Related topics:

- Configuring dynamic connections

Dynamic Display Names panel

The panel is used for managing dynamic display names for business view elements.



The options available in this panel are as follows:

Add Dynamic Display Names

Opens the Add Dynamic Display Names dialog to add dynamic display names for business view elements in a catalog for a specified SID (security identifier).

Delete

Deletes the selected dynamic display name resources.

Dynamic display name resources table

All columns are sortable except for Dynamic Display Names. To sort the table by a certain column, click on the column name. By Default the table is sorted by Catalog.

- **Checkbox**
  Specifies whether or not to select a dynamic display name resource. Select the checkbox on the column header to select all the dynamic display name resources.

The table consists of these columns:

- **Catalog**
  Displays the catalogs of the dynamic display name resources with the full resource path, for example, `/SampleReports/SampleReports.cat`.

- **Organization Name**
  Shows the organizations that the SIDs belong to. The column is available to system admin when the organization feature is enabled.

- **SID**
  Displays the security identifiers (SID). An SID can be a group, a role, or a user in the JReport Server security system.

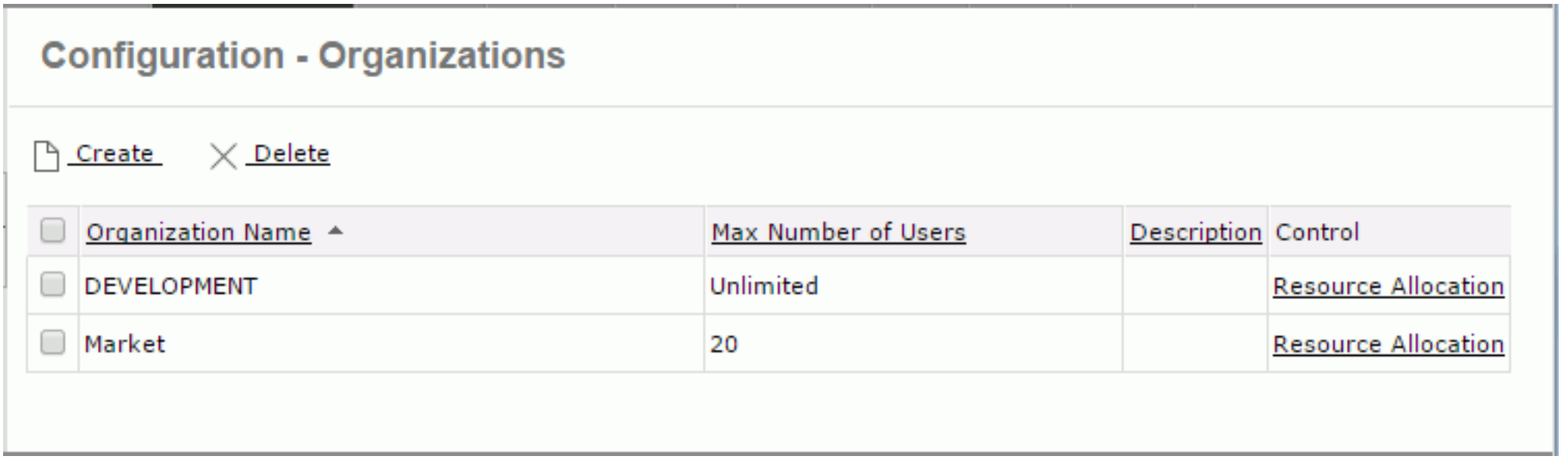
- **Dynamic Display Names**
  - **Edit**
    Opens the Edit Dynamic Display Names dialog to edit the specified dynamic display name resource.
  - **Import**
    Opens the Select Dynamic Display Name File dialog to import dynamic display names from a local resource bundle file.
  - **Export**
    Exports the dynamic display names to a resource bundle file.

Related topics:

- Managing dynamic display names of business view elements

Organizations panel

The panel is used for organizing users into different groups.



The options available in this panel are as follows:

Create

Creates an organization in the Create Organization dialog.

Delete

Deletes the selected organizations.

Organization table

- **Checkbox**
  Specifies whether or not to select an organization. Select the checkbox on the column header to select all the organizations.

The table consists of the following columns. The organizations can be sorted by any of them (to sort the table by certain column, click on the column name).

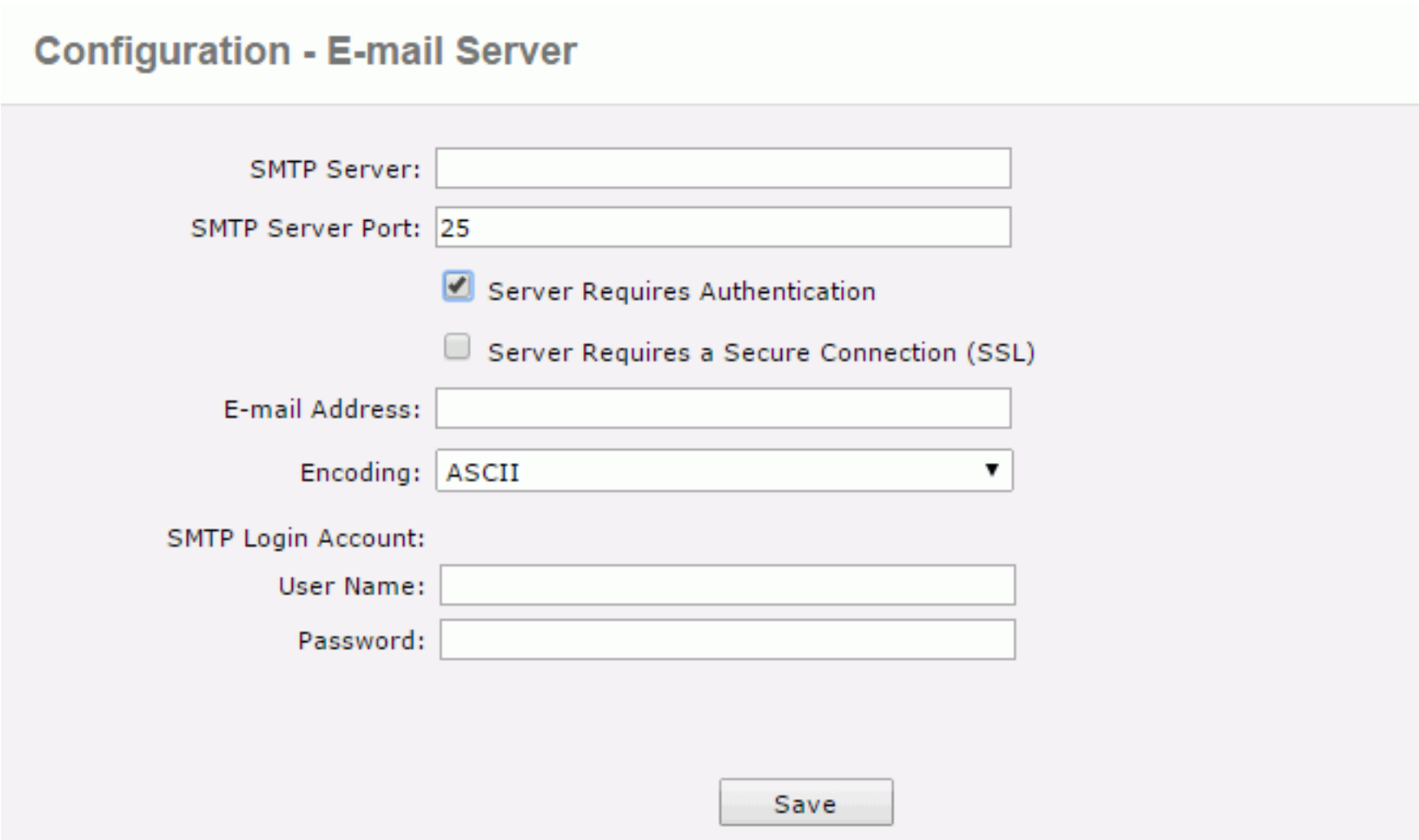
- **Organization Name**
  Displays the organization names.
- **Max Number of Users**
  Displays the maximum number of users allowed in the organizations. The values are editable. Double-click in the text box and then input a number or select one from the drop-down list.
- **Description**
  Displays the information about the organizations.
- **Control**
  - **Resource Allocation**
    Opens the Resource Allocation dialog to allocate server resources to the specific organization.

Related topics:

- Managing organization

E-mail Server panel

The panel is used for configuring the e-mail server.



The options available in this panel are as follows:

SMTP Server

Specifies the numeric or named host of the machine where the e-mail server is located.

SMTP Server Port

Specifies the port where the e-mail server runs.

Server Requires Authentication

This option must be checked if an SMTP server requires authentication.

Server Requires a Secure Connection (SSL)

This option must be checked if an SMTP server requires a secure connection (SSL).

E-mail Address

Specifies the address of the e-mail sender. You must specify an address and make sure that the format of the specified address is valid.

Encoding

Specifies the encoding for the titles and comments of notification e-mails.

SMTP Login Account

The following options are available when the option Server Requires Authentication is checked.

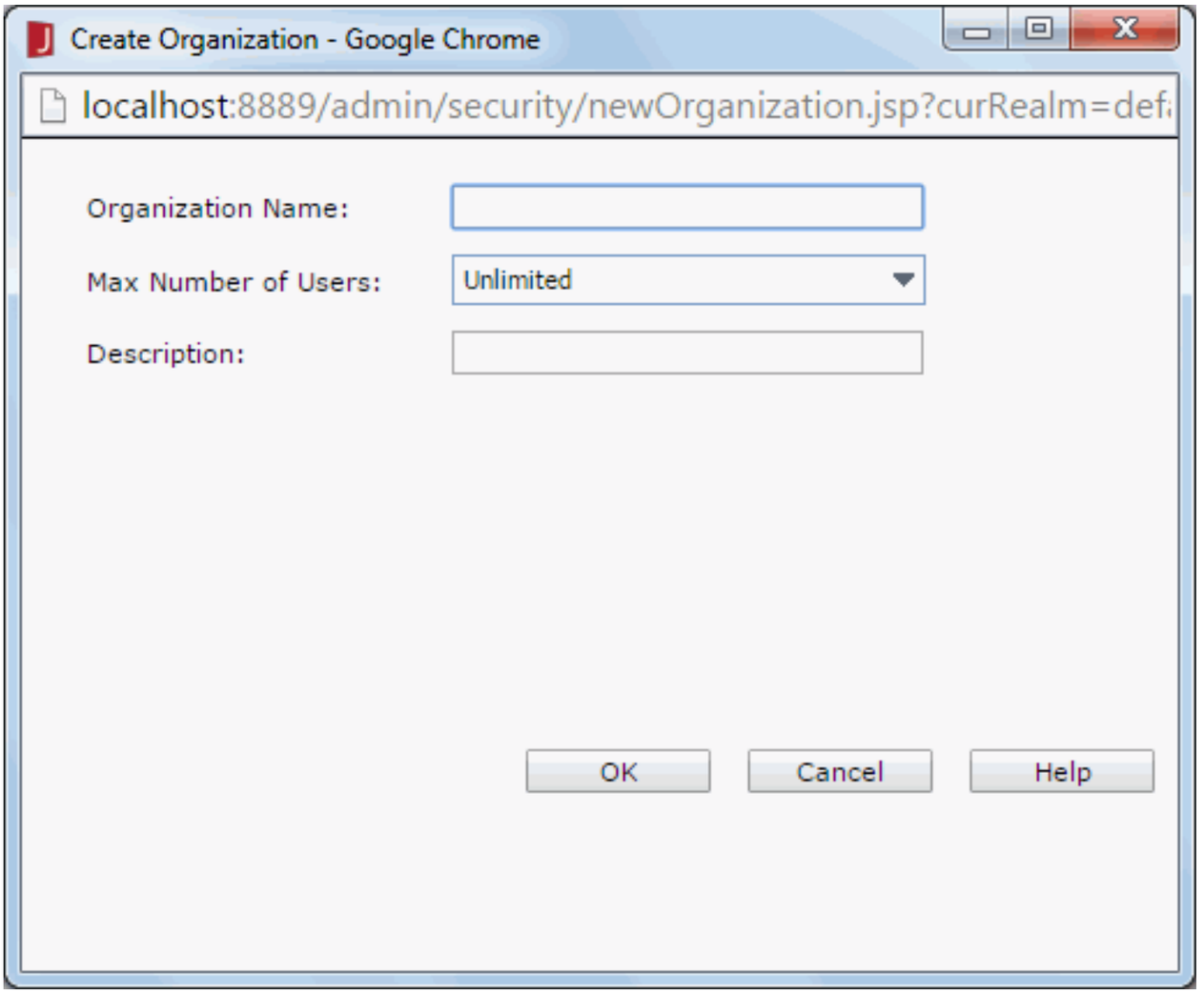
- **User Name**
  Specifies the user name for logging in the SMTP server.
- **Password**
  Specifies the password for logging in the SMTP server.

Save

Saves the settings which take effect immediately.

Create Organization dialog

The dialog appears after you click the Create link in the Organizations panel. It is used for creating an organization.



The options available in this dialog are as follows:

Organization Name

Specifies the name of the organization. The system wide organization name is *System*; therefore, it is reserved. The organization names should be distinct. A valid name can contain only '0' to '9', 'a' to 'z', 'A' to 'Z', '&', '-', '_', '.' and blank and no more than 32 characters.

Max Number of Users

Specifies the maximum number of users allowed in the organization. The default value *unlimited* means that the number is not limited. You can select a value from the drop-down list or input an integer number in the combo box directly. The **x** button in the combo box is used to clear the input text.

Description

Provides information about the organization.

OK

Creates the organization and exits the dialog.

Cancel

Does not create an organization and closes the dialog.

Help

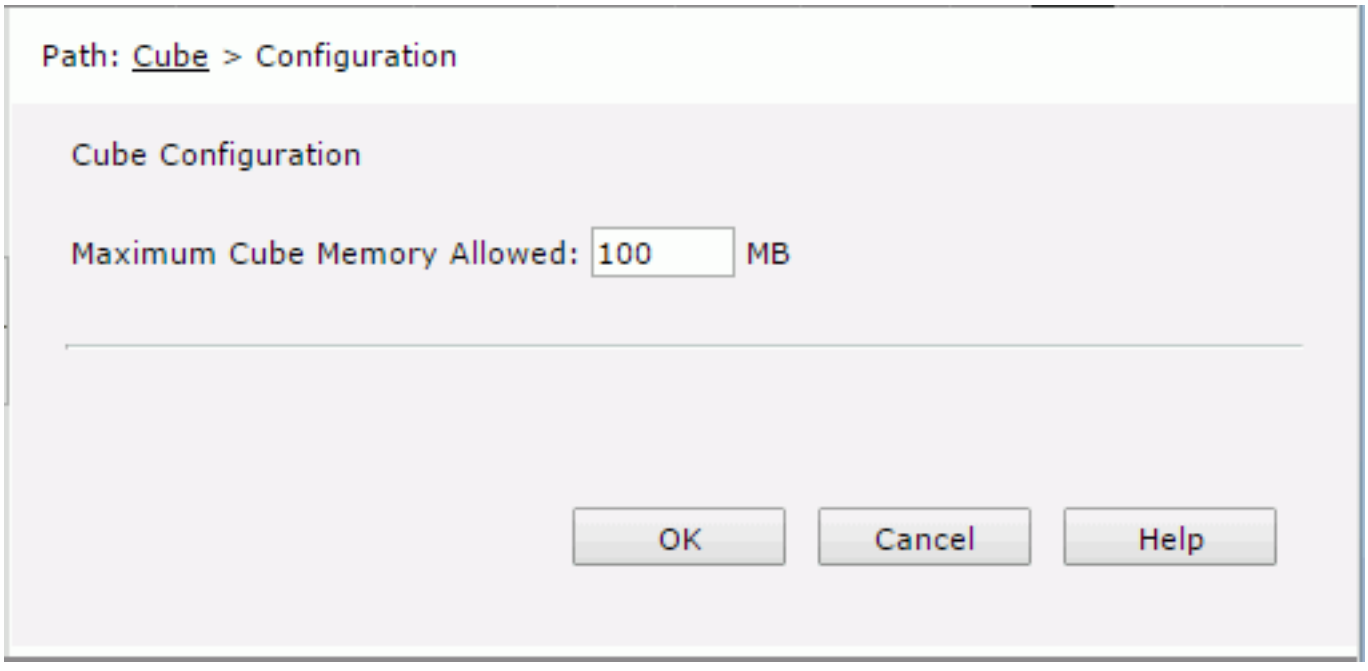
Displays the help document about this feature.

Related topics:

- Managing organizations

Cube Configuration dialog

This dialog appears when you click Configuration in the Cube dialog. This is used for specifying the allowed maximum cube memory size.



The options available in this dialog are as follows:

Maximum Cube Memory Allowed

Specifies the allowed maximum cube memory size in megabytes. The total allocated memory for each cube cannot exceed this value. It does not take into consideration the actual memory used by the in-memory or if some of the cubes are swapped to disk. This value should not be larger than 80% of your Java Heap size.

The value must be provided before the admin schedules a cube task for the first time. It can also be specified by the property "server.cube.maxmemory" in the server.properties file in the `<install_root>\bin` directory.

OK

Applies the setting.

Cancel

Resets the setting to the last saved value.

Help

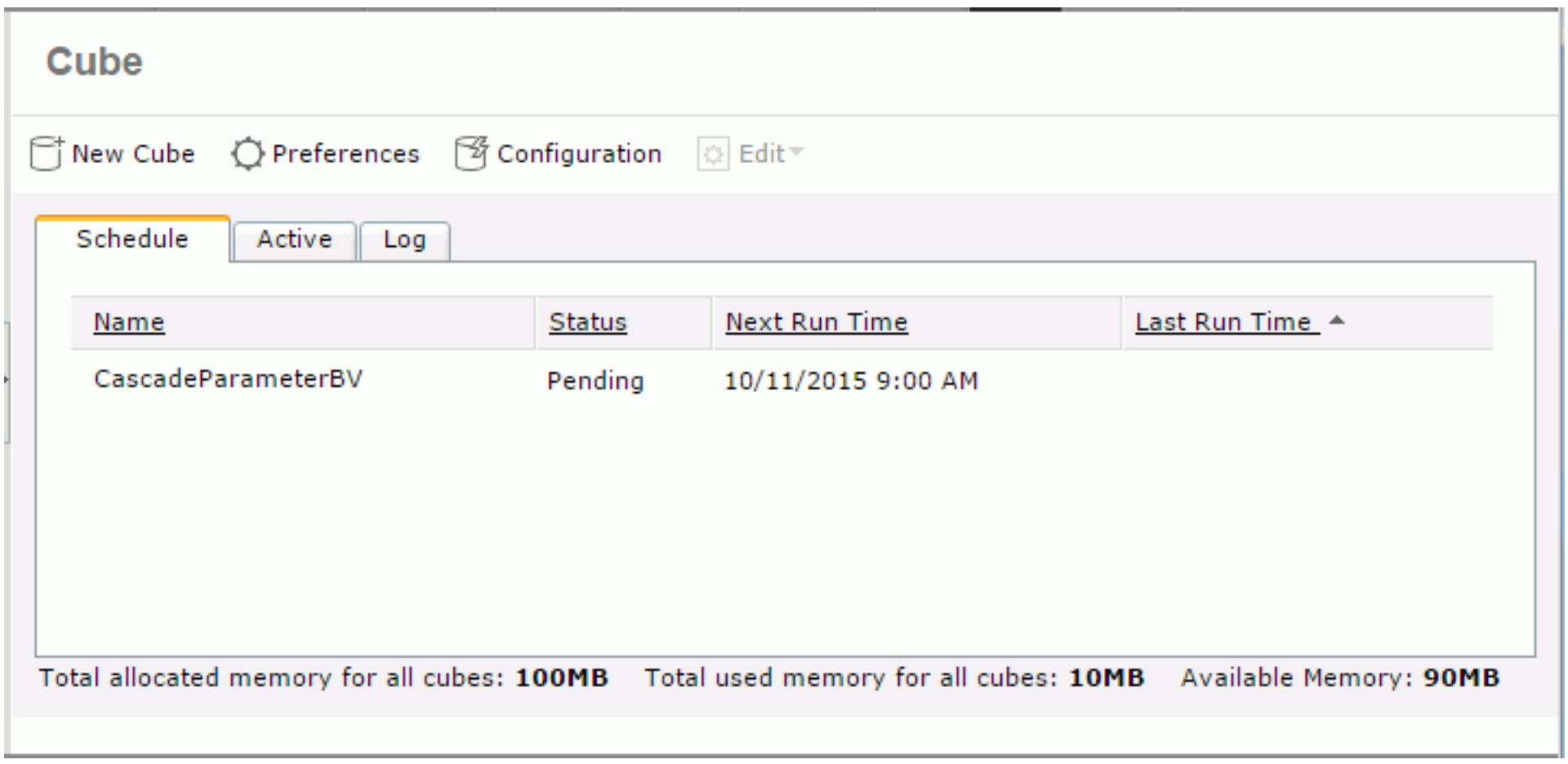
Displays the help document about this feature.

Related topics:

- Creating in-memory cubes

Cube dialog

The dialog appears after you click Cube on the system toolbar of the JReport Administration page. It helps you to create and manage in-memory cubes.



New Cube

Opens the New Cube dialog in which you can schedule a task that defines when to create in-memory cube for the specified business view.

Preferences

Specifies which columns will be displayed in the current tab.

Configuration

Opens the Cube Configuration dialog for setting the maximum cache size allowed for all in-memory cubes. Not available to organization admin.

Edit

Edits the selected cube task.

- **Enable / Disable**
  This option is enabled when one or more tasks are selected in the Schedule or Active tab. It is used to enable the disabled tasks or disable the enabled tasks. When multiple tasks are selected with some enabled and some disabled, this option shows "Enable" which enables those disabled tasks among the selected.
- **Properties**
  Edits the scheduling information of the selected cube task.
- **Delete**
  Removes the selected cube tasks from the tab.

Schedule tab

This tab displays the scheduled cube tasks.

Active tab

This tab displays the scheduled cube tasks that have generated in-memory cubes which have data and are ready for use.

Log tab

This tab displays the history information of all cube tasks.

For more information about the three tabs, refer to Viewing and managing the cube tasks.

Summary Information

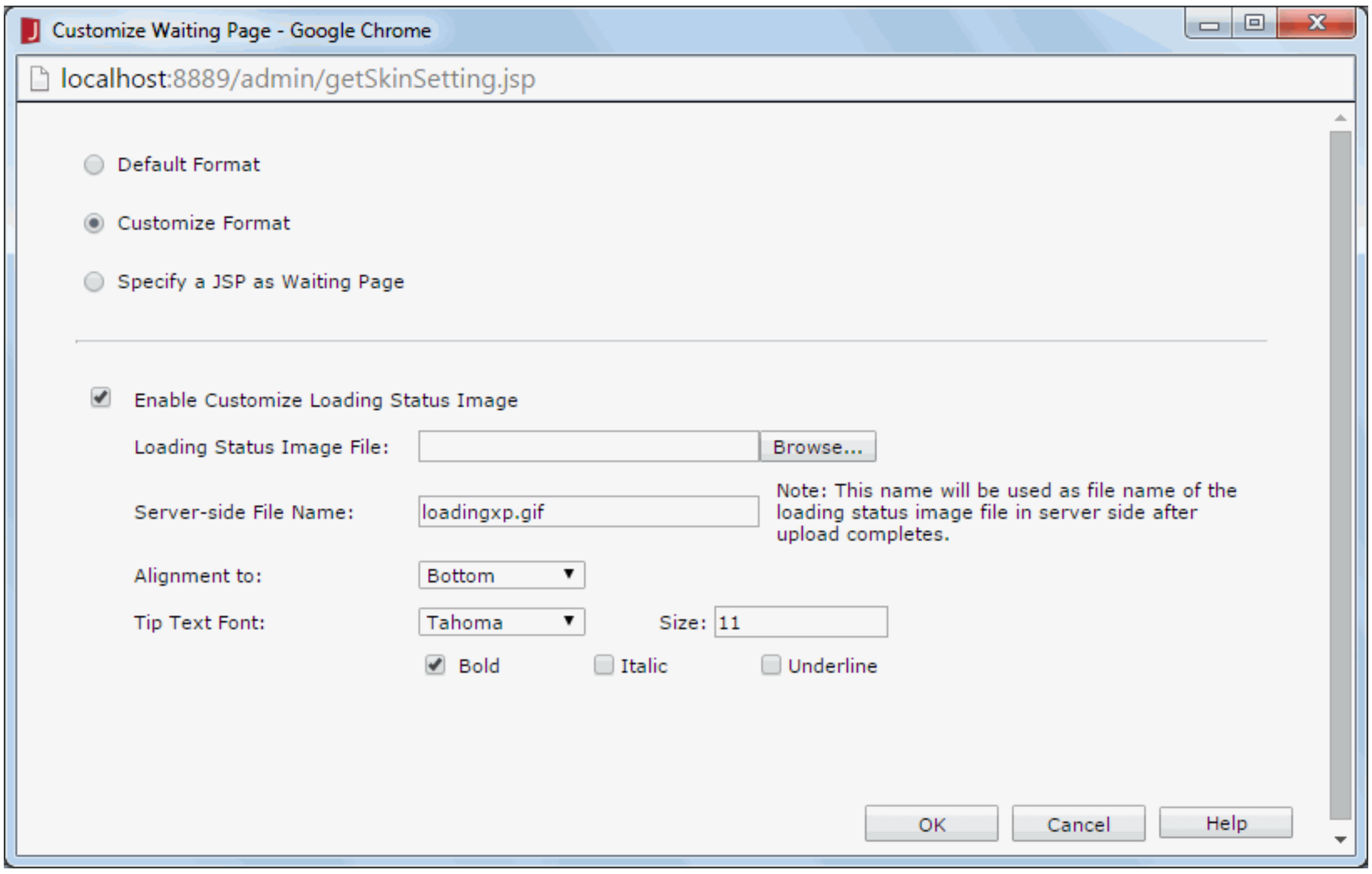
- **Total allocated memory for all cubes**
  Shows how much memory all in-memory cubes are allowed to use. This value is specified in the Cube Configuration dialog.
- **Total used memory for all cubes**
  Shows how much memory are currently taken up by all in-memory cubes.
- **Available Memory**
  Shows the memory that is available for use. This value equals the total allocated memory for all cubes minus the total used memory for all cubes.

Related topics:

- Managing in-memory cubes

Customize Waiting Page dialog

This dialog appears when you click the Customize button in the Profile > Customize Server Preferences > Advanced tab on the JReport Administration page. It helps you to customize the waiting screen according to your requirements.



Default Format

Applies the default format provided by JReport.

Customize Format

Allows to customize loading status image and tip text font.

- **Enable Customize Loading Status Image**
  Allows to upload a local image to Server as the loading status image appearing in waiting pages.
  - **Loading Status Image File**
    Uploads an image from the local disk. These types of images are supported: GIF, JPG, BMP, and PNG.
  - **Server-side File Name**
    This name will be used as the name of the image file after it is uploaded to server.
  - **Alignment to**
    Specifies the relative position of the image as compared to tip text in waiting pages.
- **Tip Text Font**
  Specifies font face and size and style of tip text in the waiting pages.

Specify a JSP as Waiting Page

Allows customized JSP files to implement the waiting pages.

- **JSP File**
  Uploads a JSP from the local disk. Note that the required JSP file is not exactly a complete JSP but a segment. Its body can contain only one root tags and the non-body section can contain only tags that are allowed in the <body></body> tags.

  The following is a sample:

```
<div>
    <table  border="0" cellspacing="0" cellpadding="0" align="center">
        <tr>
            <td valign="middle" align="center">
                <table cellpadding="5" cellspacing="0" border="0">
                    <tr>
                        <td>
                            <font
                                style="font-size:12px;font-family:Verdana;">
                                waiting....processing
                            </font>
                        </td>
                    </tr>
                    <tr>
                        <td>
                            <img src="../images/loadingdefault.gif" />
                        </td>
                    </tr>
                </table>
            </td>
        </tr>
    </table>
</div>
```

- **File Name**
  This name will be used as the name of the JSP file after it is uploaded to server.

OK

Applies the changes and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Help

Displays the help document about this feature.

Dashboard Properties dialog

You can set the properties of a dashboard in its Properties dialog according to your requirements.

The Properties dialog is divided into the following two tabs:

- General
- Permission

Reset

Discards your modifications and restores the dialog to its default status.

OK

Retains the settings and submits the task to server.

Cancel

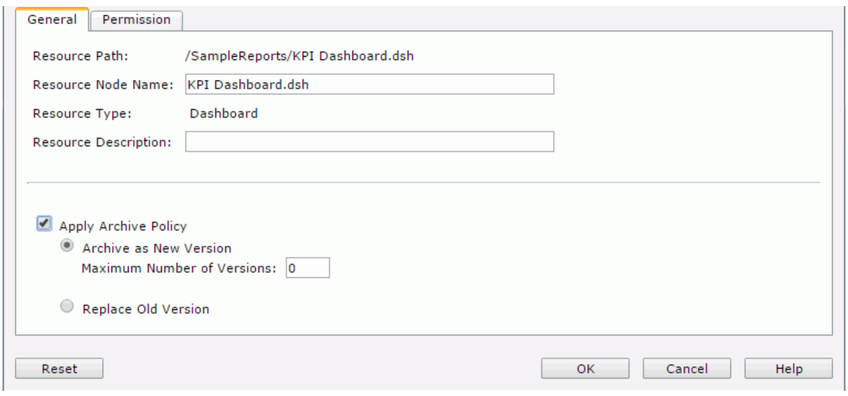
Cancels any settings and closes the dialog.

Help

Displays the help document about this feature.

General tab

Specifies the general properties of the dashboard.



Resource Path

Shows the resource path.

Resource Node Name

Specifies the name for the dashboard.

Resource Type

Shows the type of the resource.

Resource Description

Specifies the description for the dashboard.

National Language Support

Specifies whether to enable the NLS feature for the dashboard. This option is available when the Properties dialog is opened from the JReport Administration page.

[Custom Field Name]

Specifies value of the custom field for the dashboard. A custom field can be regarded as a resource property and is available when it is enabled.

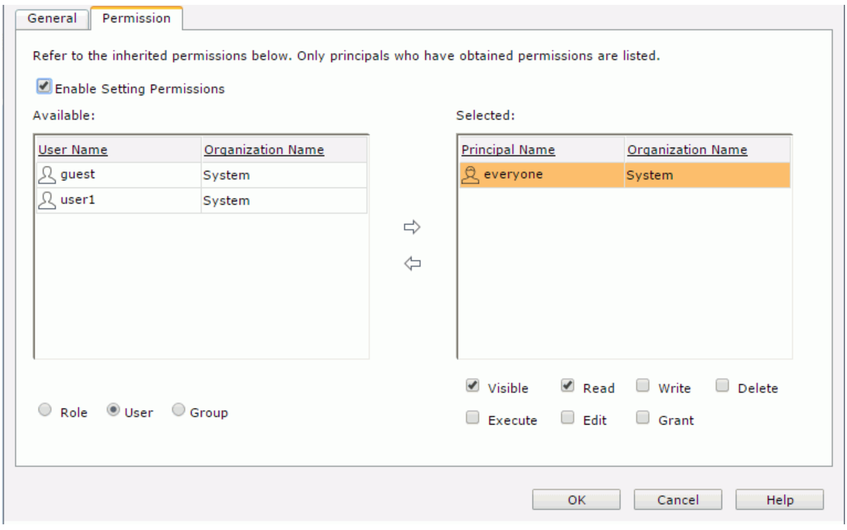
Apply Archive Policy

Applies an archive policy to the dashboard versions.

- **Archive as New Version**
  Specifies whether to use multiple versions for the dashboard.
  - **Maximum Number of Versions**
    Specifies the maximum number of versions that will be listed in the version table of the dashboard. The default value is 0, which means that the version number is unlimited.
- **Replace Old Version**
  Specifies to replace the old version when a new version is generated.

Permission tab

Specifies permissions of roles/users/groups on the dashboard. This tab is available when the dashboard is in the Public Reports folder.



Enable Setting Permissions

Enables the setting of permissions.

Available

Lists the roles/users/groups to which you can assign permissions. When organization is enabled, the Organization Name column will be available in the table.

- **Role**
  If checked, all roles will be displayed in the Available box for you to assign permissions.
- **User**
  If checked, all users will be displayed in the Available box for you to assign permissions.
- **Group**
  If checked, all groups will be displayed in the Available box for you to assign permissions.

Adds the selected role, user or group to the Selected box.

Removes the selected role, user or group from the Selected box.

Selected

Select a role/user/group in the Selected box and then select the permissions you would like the role/user/group to have on the dashboard. For what each permission is used for, see Permission.

When organization is enabled, the Organization Name column will be available in the table.

Related topics:

- Changing resource and folder properties

Data dialog

After you click Data on the system toolbar of the JReport Administration page, the entries to the following panels are available on the drop-down menu. The panels help you to configure your databases.

Note: The Profiling DB option may not be shown on the drop-down menu. To use the profiling database, you need to set the server.profiling.enable property to true in the server.properties file in the `<install_root>\bin` directory.

- System DB panel
  It allows you to configure your system database and backup server data. The system database holds resources of the global server scope, such as server.properties, global NLS, and so on.
- Realm DB panel
  It allows you to manage server data, such as configuring your realm database, backing up server data, archiving/restoring server data, viewing backup and archive files' summary information, and clearing invalid resource nodes. The realm database holds information of folders, nodes, versions, the security system, schedule, and the completed table.
- Profiling DB panel
  It allows you to manage server data, such as configuring your profiling database, backing up server data, archiving/restoring server data, and viewing backup and archive files' summary information.

System DB panel

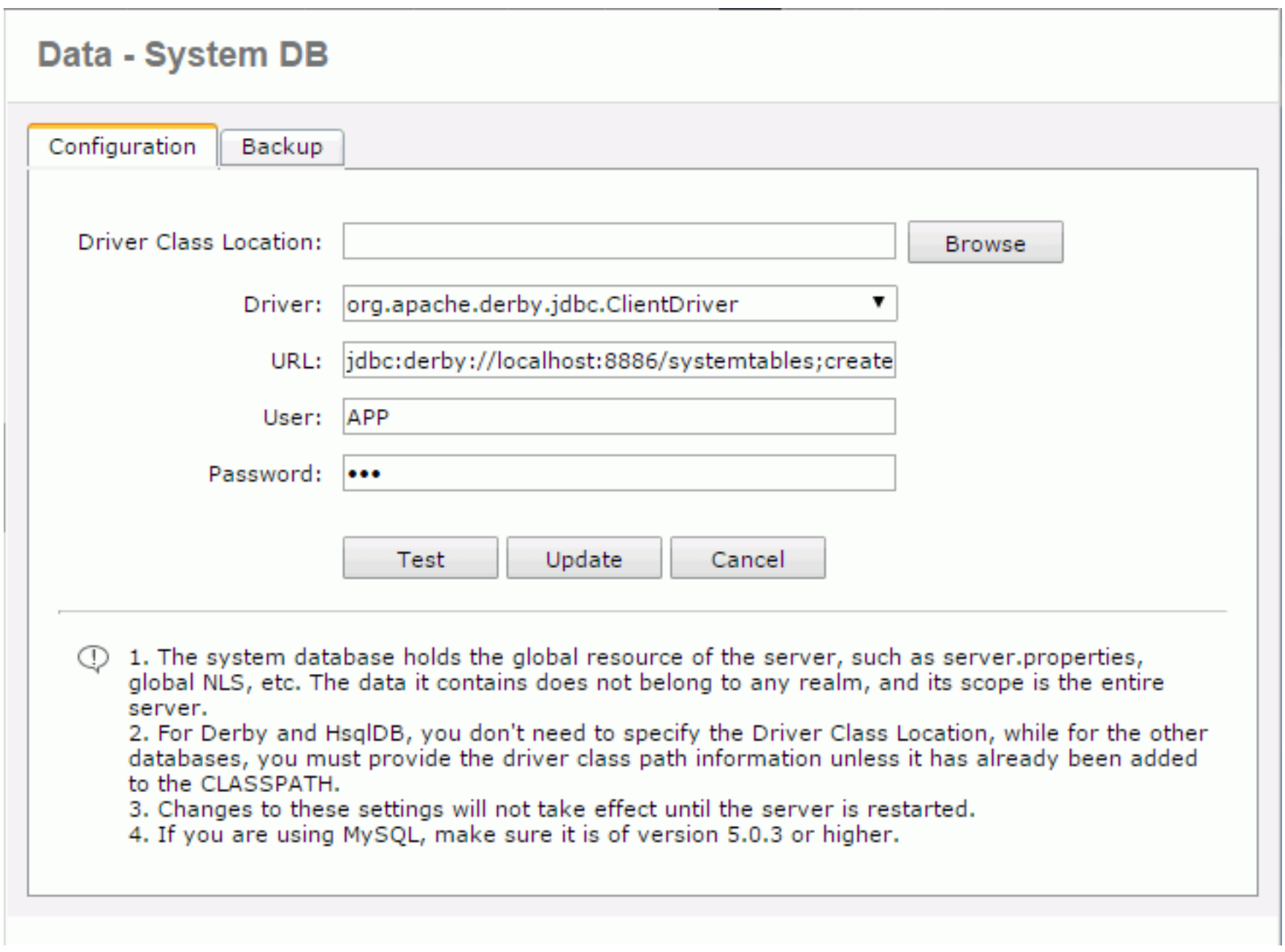
This panel is displayed when you click Data on the system toolbar of the JReport Administration page and then select System DB from the drop-down menu. It allows you to configure your system database and backup server data.

The following are tabs contained in the panel:

- Configuration tab
- Backup tab

Configuration tab

This tab allows you to configure your system database to store server data.



Driver Class Location

Specifies the path of the JDBC driver. It can be null, which means that JReport Server will load the driver from the VM's class path. Click the Browse button to specify the location for the driver class.

Driver

Specifies the name of the JDBC driver.

URL

Specifies a valid URL that can be used to establish a connection to your database. For the valid format of the URL, refer to the handbook of the specific driver.

User

Specifies the JDBC user name for connecting to the database.

Password

Specifies the user password.

Test

Tests whether the JDBC configuration works.

Update

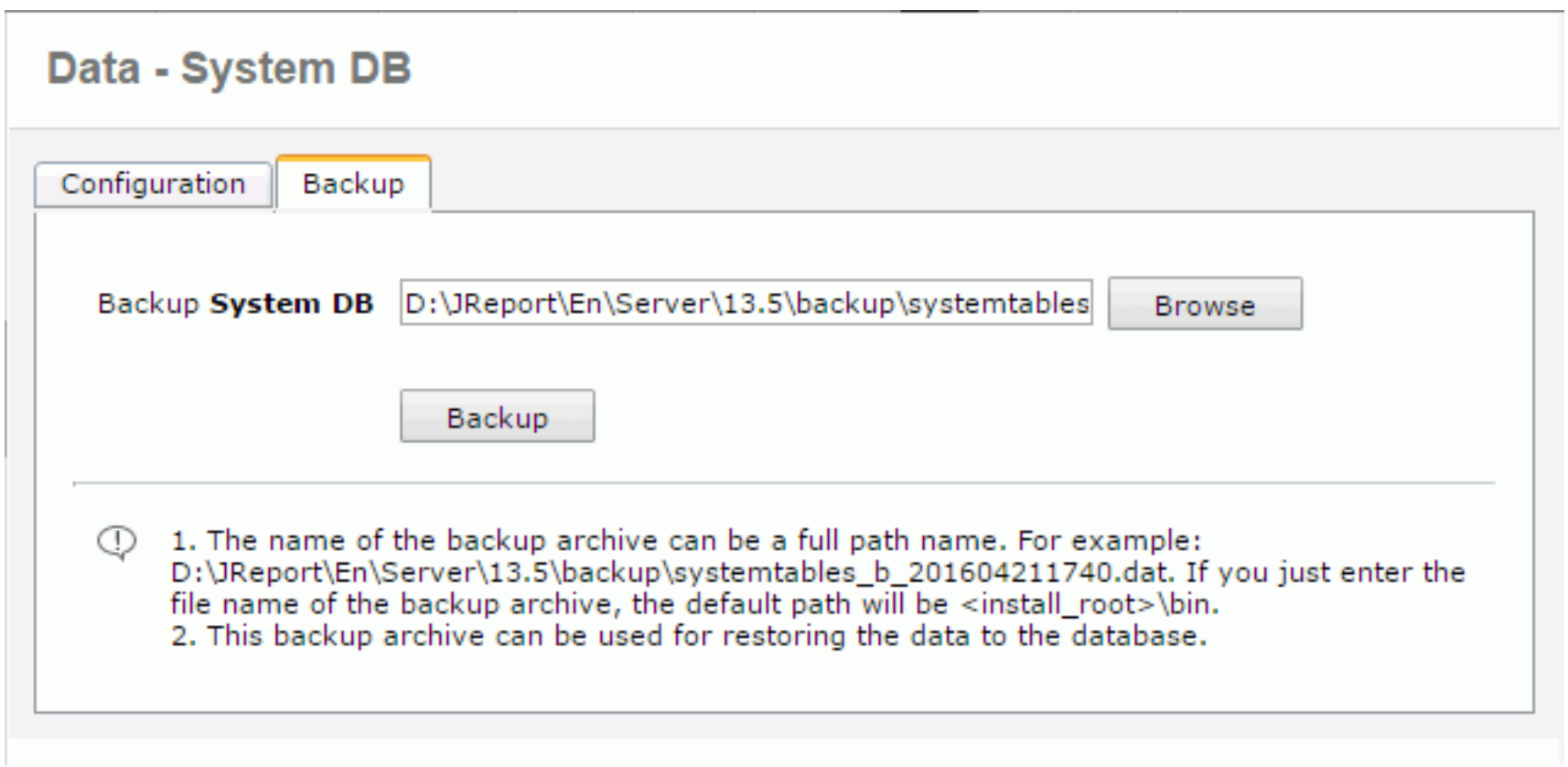
Updates the JDBC configuration and saves it to disk.

Cancel

Cancels the settings.

Backup tab

This tab allows you to back up your system database for archiving.



Backup System DB

Backups JReport Server's system database to an archive. Click the Browse button to specify a file path to store the backup archive.

Backup

Begins the backup process.

Related topics:

- Managing server data

Realm DB panel

The panel is displayed when you click Data on the system toolbar of the JReport Administration page and then select Realm DB from the drop-down menu. It allows you to manage server data, such as configuring your realm database, backing up server data, archiving/restoring server data, viewing backup and archive files' summary information, and clearing invalid resource nodes.

The following are tabs contained in the panel:

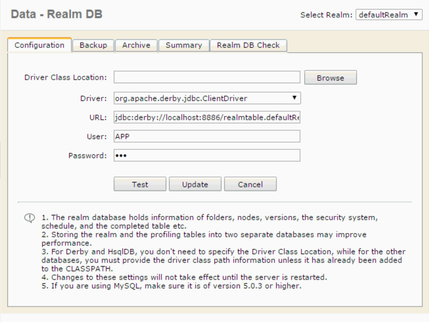
- Configuration tab
- Backup tab
- Archive tab
- Summary tab
- Realm DB Check tab

Select Realm

Before you can manage the realm database, you need to first select a realm from the drop-down list.

Configuration tab

This tab allows you to configure your realm database to store server data.



Driver Class Location

Specifies the path of the JDBC driver. It can be null, which means that JReport Server will load the driver from the VM's class path. Click the Browse button to specify the location for the driver class.

Driver

Specifies the name of the JDBC driver.

URL

Specifies a valid URL that can be used to establish a connection to your database. For the valid format of the URL, refer to the handbook of the specific driver.

User

Specifies the JDBC user name for connecting to the database.

Password

Specifies the user password.

Test

Tests whether the JDBC configuration works.

Update

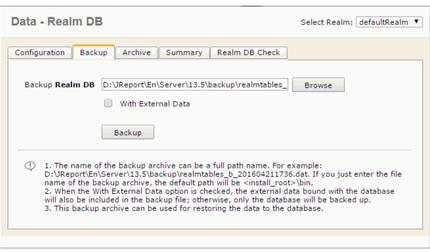
Updates the JDBC configuration and saves it to disk.

Cancel

Cancels the settings.

Backup tab

This tab allows you to back up your system's database for archiving.



Backup Realm DB

Backups JReport Server's realm database to an archive. Click the Browse button to specify a file path to store the backup archive.

With External Data

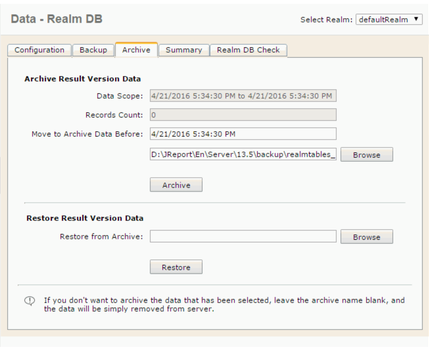
If it is checked, the external data bound with the database will also be archived. Otherwise, only the database will be archived.

Backup

Begins the backup process.

Archive tab

This tab allows you to archive the result version data in an archive file.



Archive Result Version Data

- **Data Scope**
  Displays the result version data range.
- **Records Count**
  Displays the number of sets of result version data in the database.
- **Move to Archive Data Before**
  Specifies the range of data that you want to archive. The range is defined using date, for example, you can archive data before a certain date. Click the Browse button to specify the file path for storing the archived data.
- **Archive**
  Begins the archiving process.

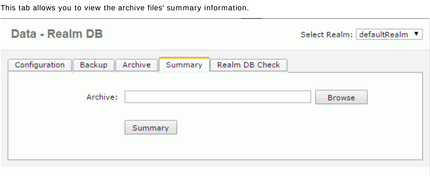
Restore Result Version Data

After data has been saved to an archive file, you then have the option to restore it.

- **Restore from Archive**
  Specifies the archive file that contains the archived result version data. Click the Browse button to specify the archive file.
- **Restore**
  Begins the restoring process.

Summary tab

This tab allows you to view the archive files' summary information.



Archive

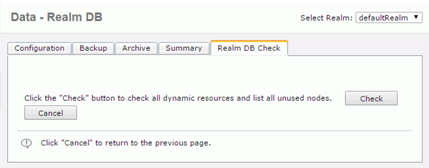
Specifies the name of the archive from which you want to retrieve summary information. Click the Browse button to specify the archive.

Summary

Begins the summary process and returns summary information about the specified archive.

Realm DB Check tab

This tab allows you to clear invalid resource nodes from JReport Server.



Check

Checks whether there are any resource nodes that have lost external data. If any corresponding files (the resource's external data) have been manually deleted, the resource nodes will become invalid. Invalid nodes should be cleared from JReport Server.

Invalid Dynamic Nodes

Displays all invalid resource nodes in the database.

Delete

Deletes the specified invalid resource nodes.

Cancel

Cancels the operation.

Related topics:

- Managing server data

Profiling DB panel

This panel is displayed when you click Data on the system toolbar of the JReport Administration page and then select Profiling DB from the drop-down menu. It allows you to manage server data, such as configuring your profiling database, backing up server data, archiving/restoring server data, and viewing backup and archive files' summary information.

The following are tabs contained in the panel:

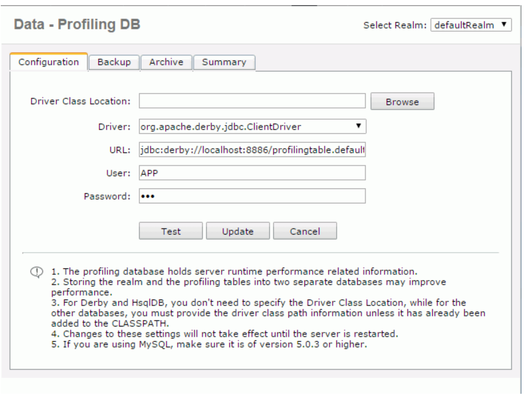
- Configuration tab
- Backup tab
- Archive tab
- Summary tab

Select Realm

Before you can manage the profile database, you need to first select a realm from the drop-down list.

Configuration tab

This tab allows you to configure your profiling database to store server data.



Driver Class Location

Specifies the path of the JDBC driver. It can be null, which means that JReport Server will load the driver from the VM's class path. Click the Browse button to specify the location for the driver class.

Driver

Specifies the name of the JDBC driver.

URL

Specifies a valid URL that can be used to establish a connection to your database. For the valid format of the URL, refer to the handbook of the specific driver.

User

Specifies the JDBC user name for connecting to the database.

Password

Specifies the user password.

Test

Tests whether the JDBC configuration works.

Update

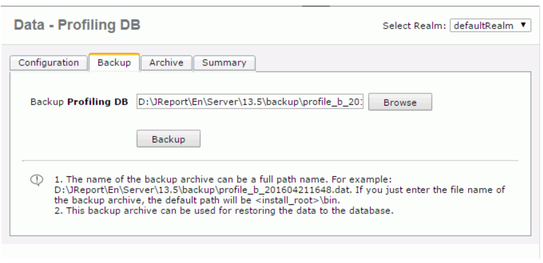
Updates the JDBC configuration and saves it to disk.

Cancel

Cancels the settings.

Backup tab

This tab allows you to back up your system's database for archiving.



Backup Profiling DB

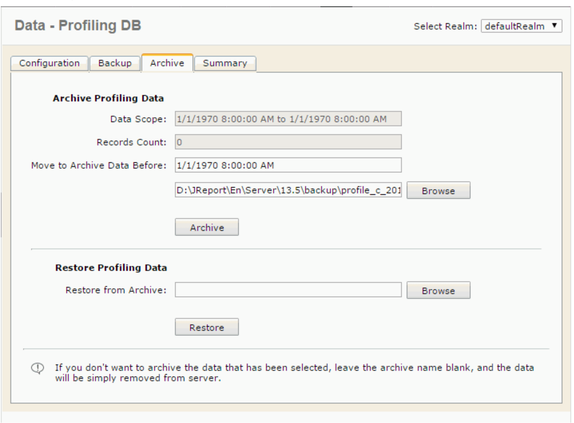
Backups JReport Server's profiling database to an archive. Click the Browse button to specify a file path to store the backup archive.

Backup

Begins the backup process.

Archive tab

This tab allows you to archive the result version data in an archive file.



Archive Profiling Data

- **Data Scope**
  Displays the result version data range.
- **Records Count**
  Displays the number of sets of result version data in the database.
- **Move to Archive Data Before**
  Specifies the range of data that you want to archive. The range is defined using date, for example, you can archive data before a certain date. Click the Browse button to specify the file path for storing the archived data.
- **Archive**
  Begins the archiving process.

Restore Profiling Data

After data has been saved to an archive file, you then have the option to restore it.

- **Restore from Archive**
  Specifies the archive file that contains the archived result version data. Click the Browse button to specify the archive file.
- **Restore**
  Begins the restoring process.

Summary tab

This tab allows you to view the archive files' summary information.

Archive

Specifies the name of the archive from which you want to retrieve summary information. Click the Browse button to specify the archive.

Summary

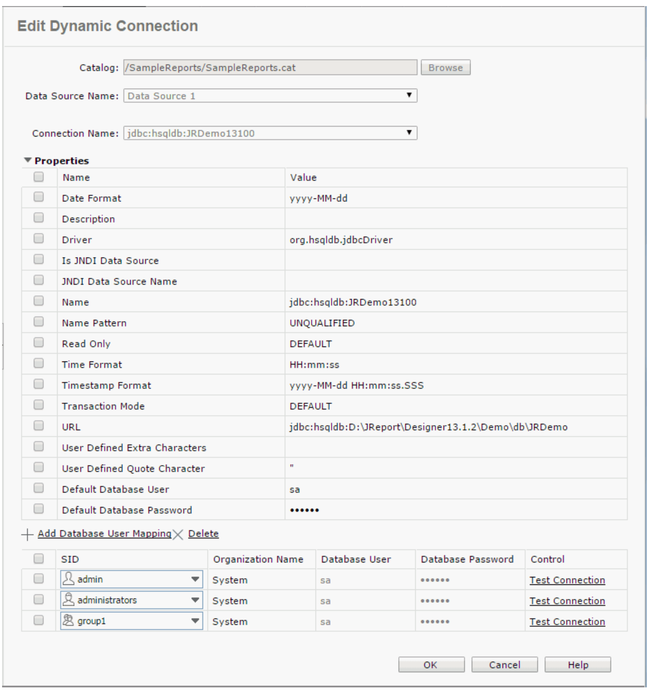
Begins the summary process and returns summary information about the specified archive.

Related topics:

- Managing server data

Edit Dynamic Connection dialog

The dialog appears when you click a connection name in the Connection Name column in the Dynamic Connections panel. It helps you to edit the selected dynamic connection.



Catalog

Displays the catalog of the dynamic connection. It is read only.

Data Source Name

Displays the data source of the dynamic connection. It is read only.

Connection Name

Specifies a connection to connect to the data source from the drop-down list. It is read only.

Properties

Displays the properties of the database connection. You can click it to expand or collapse the property table.

Here you can change some properties of the database connection. The table shows the values of the original connection properties initially.

- **Checkbox**
  After you select the checkbox for a property, you will be able to change the property value. Select the checkbox on the column header and the checkboxes for all the properties will be selected. Initially all the properties are unchecked and uneditable. Whenever a property is unchecked, it always shows the original value regardless of whether you have changed the value or not.
- **Name**
  Displays the property names.
- **Value**
  Specifies the property values.

There are the following properties:

- **Date Format**
  Specifies the default Date format corresponding to the database.
- **Description**
  Specifies the description of the connection.
- **Driver**
  Specifies the class name of the JDBC driver such as oracle.jdbc.driver.OracleDriver.
- **Is JNDI Data Source**
  Specifies whether it is a JNDI data source.
- **JNDI Data Source Name**
  Specifies the JNDI data source name when the Is JNDI Data Source property is true.
- **Name**
  Specifies the name of the connection.
- **Name Pattern**
  Specifies whether or not catalog or schema is used in data manipulation. Choose an option from the drop-down list.
  - **Unqualified** - Neither catalog nor schema is included in data manipulation. Example: SELECT t.c FROM t
  - **Qualified_2part** - Uses schema in data manipulation. Example: SELECT schema.t.c FROM schema.t
  - **Qualified_3part**- Uses both catalog and schema in data manipulation. Example: SELECT catalog.schema.t.c from catalog.schema.t
- **Read Only**
  Specifies the mode to open the connection to the JDBC data source. The initial setting is Default which uses the mode specified by the DBMS Administrator which could be read only or read & write. Choose an option from the drop-down list.
  - **Default** - May be read & write or read only depending on the DBMS default setting.
  - **Read only** - Allows the driver to optimize performance for reporting which does not need to write to the DBMS.
  - **Read & write** - Opens the DBMS with updates enabled which requires more processing to ensure concurrency control.
- **Time Format**
  Specifies the default time format corresponding to the database.
- **Timestamp Format**
  Specifies the default timestamp format corresponding to the database.
- **Transaction Mode**
  Specifies the transaction mode for the connection. Choose an option from the drop-down list.
  - **Default** - Indicates the transaction information cannot be got from JDBC connection.
  - **None** - Indicates that transactions are not supported.
  - **Read_uncommitted** - Dirty reads, non-repeatable reads and phantom reads can occur. This mode will speed up the transaction of the catalog.
  - **Read_committed** - Dirty reads are prevented; non-repeatable reads and phantom reads can occur.
  - **Repeatable_read** - Dirty reads and non-repeatable reads are prevented; phantom reads can occur.
  - **Serializable** - Dirty reads, non-repeatable reads and phantom reads are prevented.
- **URL**
  Specifies the JDBC URL which establishes the connection to the database, for example, jdbc:oracle:thin:@localhost:1521:ORCL.
- **User Defined Extra Characters**
  Specifies extra characters of user defined quote qualifier.
- **User Defined Quote Characters**
  Specifies the quote character of user defined quote qualifier.
- **Default Database User**
  Specifies the user name for connecting to the database, which is determined by the database DBA. Null means using the default database user name.
- **Default Database Password**
  Specifies the password for connecting with the database, which is determined by the database DBA. Null means using the default database password.

Add Database User Mapping

Adds a new database user mapping. Click to create a new line at the end of the table.

Delete

Deletes the selected database user mappings.

Database user mapping table

Checkboxes are used to specify whether or not to select the database user mapping. Select the checkbox on the column header to select all database user mappings. After you select the database user mappings, you can then delete them if you do not want them.

- **SID**
  Specifies the security identifier (SID). A SID can be a group, role or user in the JReport Server security system. You can select a value from the drop-down list or input the value in the text box. The **x** in the text box is used to clear the input text. You can define at most one database user mapping for an SID within a dynamic connection.
- **Organization Name**
  Double-click in the text box and then select an organization from the drop-down list. You can first specify the organization and then the SID. The column is available to system admin when the organization feature is enabled.
- **Database User**
  Specifies the database user name. Null means using the default database user name.
- **Database Password**
  Specifies the database password. Passwords are masked.
- **Control**
  - **Test Connection**
    Tests whether the connection configuration works by using the database use rname and password.

OK

Creates the dynamic connection and exits the dialog.

Cancel

Cancels the creation of a dynamic connection and closes the dialog.

Help

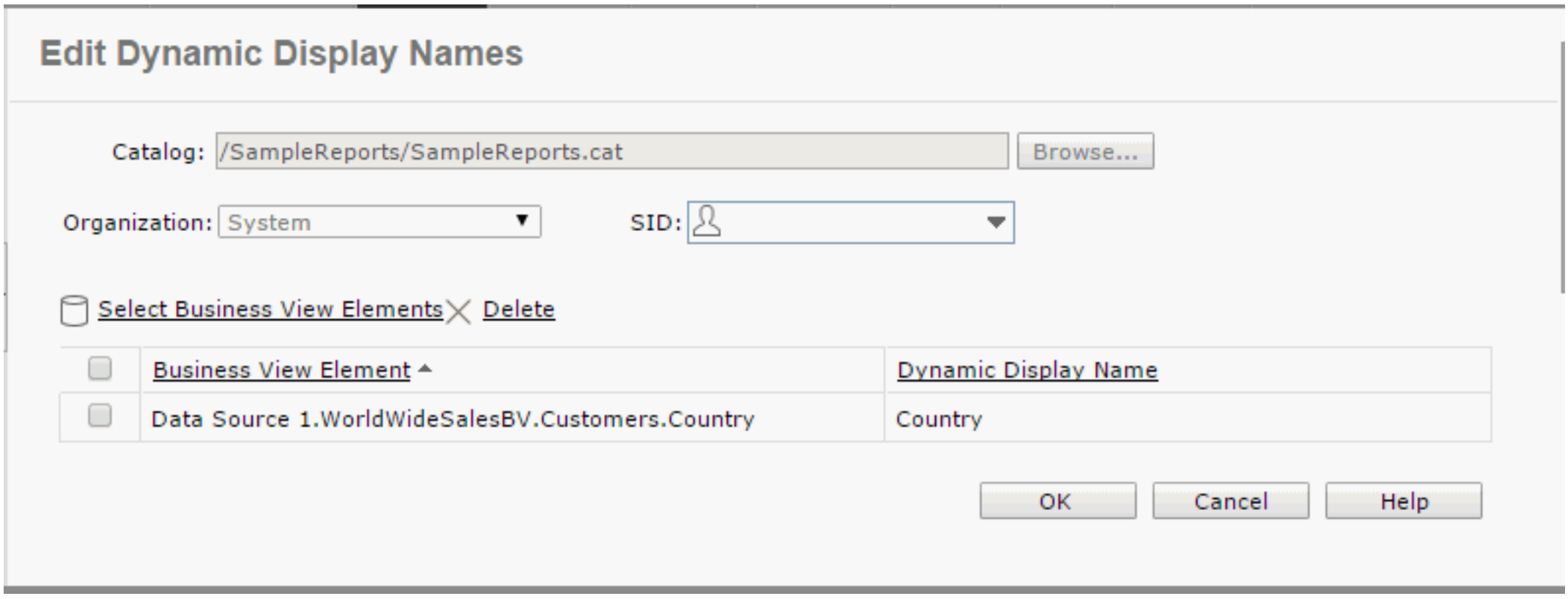
Displays the help document about this feature.

Related topics:

- Configuring dynamic connections

Edit Dynamic Display Names dialog

This dialog appears when you click the Edit link in the Dynamic Display Names column in the Dynamic Display Names panel. It helps you to edit the dynamic display names for business view elements in a catalog.



Catalog

Displays the catalog with the resource path that contains the required business views for which the dynamic display names are defined.

Organization

Displays the organization. The option is available when the organization feature is enabled.

SID

Specifies the SID for which you want to define dynamic display names. An SID can be a group, role, or user in the JReport Server security system. Select a value from the drop-down list. The **x** button in the text box is used to clear the current value. Empty means all the users (when the organization feature is enabled, empty means all the users in the specified organization).

Select Business View Elements

Opens the Select Business View Elements dialog to select business view elements in the current catalog for which to define dynamic display names.

Delete

Deletes the selected business view elements.

Business view elements table

Checkboxes are used to specify whether or not to select the business view elements. Select the checkbox on the column header to select all the business view elements. After you select the business view elements, you can then delete them if you do not want them. You can also click the underlined column header name to sort the elements either by their qualified names or display names.

- **Business View Element**
  Displays the qualified names of the business view elements with the path in the catalog.
- **Dynamic Display Name**
  Shows the display names of the business view elements. The display names are editable. Click or double-click in a name text box to enter the editing mode and then change the name. The **x** in the text box is used to clear the text.

OK

Applies the dynamic display names and exits the dialog.

Cancel

Cancels editing the dynamic display names and closes the dialog.

Help

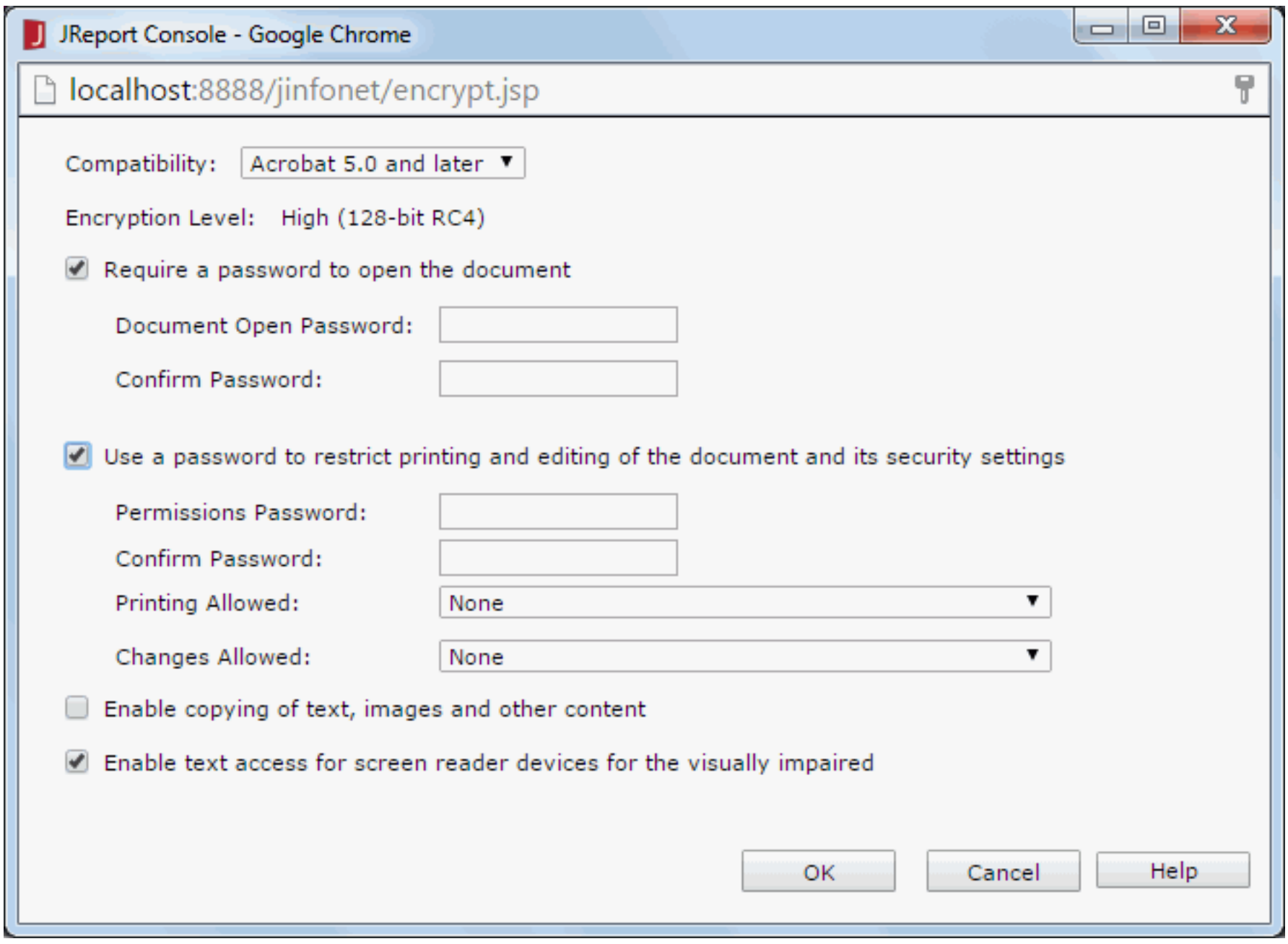
Displays the help document about this feature.

Related topics:

- Managing dynamic display names of business view elements

Encrypt dialog

This dialog appears when you click the Setting button next to the Encrypt option while configuring settings for advanced running/publishing a report in PDF format.



The following are details about the options in the dialog:

Compatibility

Specifies the encryption type from the two types that are provided here in order to encrypt a PDF document. The option Acrobat 3.0 and later uses a low encryption level (40-bit RC4), while the other option Acrobat 5.0 and later uses a high encryption level (128-bit RC4).

Encryption Level

Shows the level of the encryption compatibility that you specified in the Compatibility drop-down list.

Require a password to open the document

Specifies the Document Open Password to prevent others from opening the document without authorization.

- **Document Open Password**
  Specifies the password to prevent others from opening the document without authorization.
- **Confirm Password**
  Confirms the password you have specified in the Document Open Password text box.

Use a password to restrict printing and editing of the document and its security settings

Specifies the Permission Password to prevent others from printing and editing the document. The password you specify here cannot be the same as the one that you use to open the document.

- **Permissions Password**
  Specifies the password to prevent others from printing and editing.
- **Confirm Password**
  Confirms the password you have specified in the Permissions Password text box.
- **Printing Allowed**
  Specifies the printing quality for the PDF document.
- **Changes Allowed**
  Specifies the editing actions that are permitted in the PDF document.

Enable copying of text, images and other content

Allows others to select and copy the contents of the PDF document.

Enable text access for screen reader devices for the visually impaired

Allows visually impaired users to read the PDF document with window readers. This option is available only if the Compatibility option is set to Acrobat 5.0 or later Acrobat version.

OK

Saves the settings and exits the dialog.

Cancel

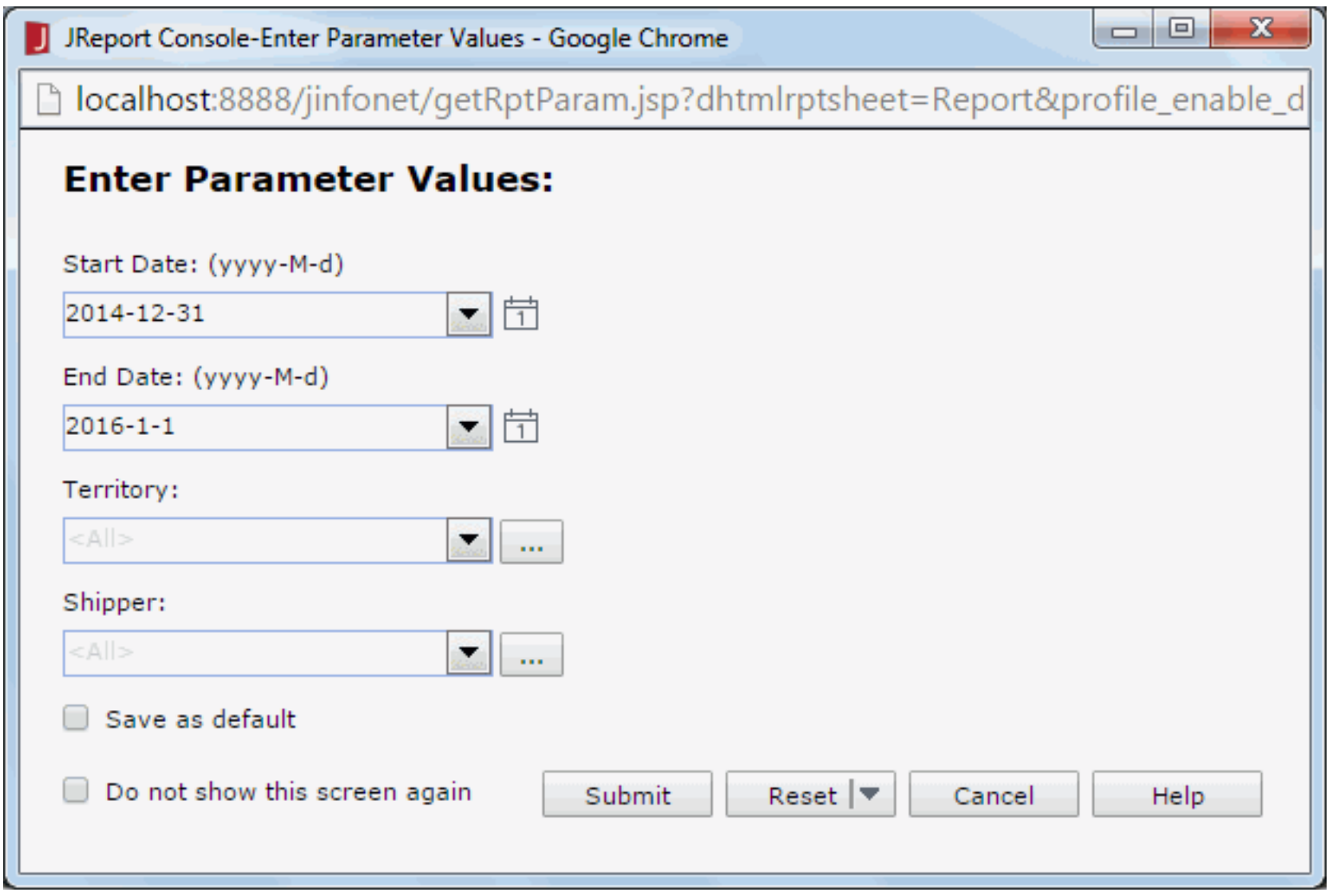
Cancels the settings and closes the dialog.

Help

Displays the help document about this feature.

Enter Parameter Values dialog

This dialog appears when you run a report with parameters. It allows you to specify the parameter values to run the report.



Parameters

Displays parameters of the report with your last-time saved default values, which could be the values saved in this dialog last time, or in the Parameter Settings dialog, or when advanced running or scheduling the report, and for a web report, could also be the values saved in the Parameters panel. Edit the values according to your requirement. You may specify parameter values in these ways.

If you have not yet set the default values on the server, or if you did but your saved default values cannot fully match the current parameters, all the parameters will use their default values specified in the parameters' definition as the initial values.

Use Saved Values

If it is available, you can select the previously saved parameter values to apply to the report and save parameter values for reuse later.

Save as default

Saves current parameter values as the default values for the report. It is a user-report level setting.

This option is an action and it takes effect after you click Submit in the dialog. Its initial status is always unselected.

For page reports, this option is available when Enable Setting Default Parameter Values For Page Report is checked in the Profile > Customize Server Preferences > Advanced tab.

For web reports, this option is available when Enable Setting Default Parameter Values For Web Report is checked in the Profile > Customize Server Preferences > Advanced tab.

Do not show this screen again

When this option is selected, the next time the report runs via operation on the console, it will use the last-time saved default parameter values directly without popping up this dialog. However, if the last-time saved default values cannot completely match the current report parameters, the dialog will still be displayed. It is a user-report level setting.

For page reports, this option is available when Enable Hiding Initial Parameter Dialog For Page Report is checked in the Profile > Customize Server Preferences > Advanced tab.

For web reports, this option is available when Enable Hiding Initial Parameter Dialog For Web Report is checked in the Profile > Customize Server Preferences > Advanced tab.

Submit

Closes this dialog and applies the specified values to run the report.

Reset

When Save as default is available in the dialog, the Reset button contains a text part and a triangle icon. You can choose to reset the values to either of the following by clicking the triangle. If you click the text part of the button directly, the values will be reset to the original default values.

- **Original Default Values**
  The default values defined in the parameters' definition.
- **User Defined Default Values**
  Your last-time saved default values.

When Save as default is unavailable in the dialog, the Reset button contains only the text part. Clicking it will reset the values to the original default values.

Cancel

Cancels running the report and closes this dialog.

Help

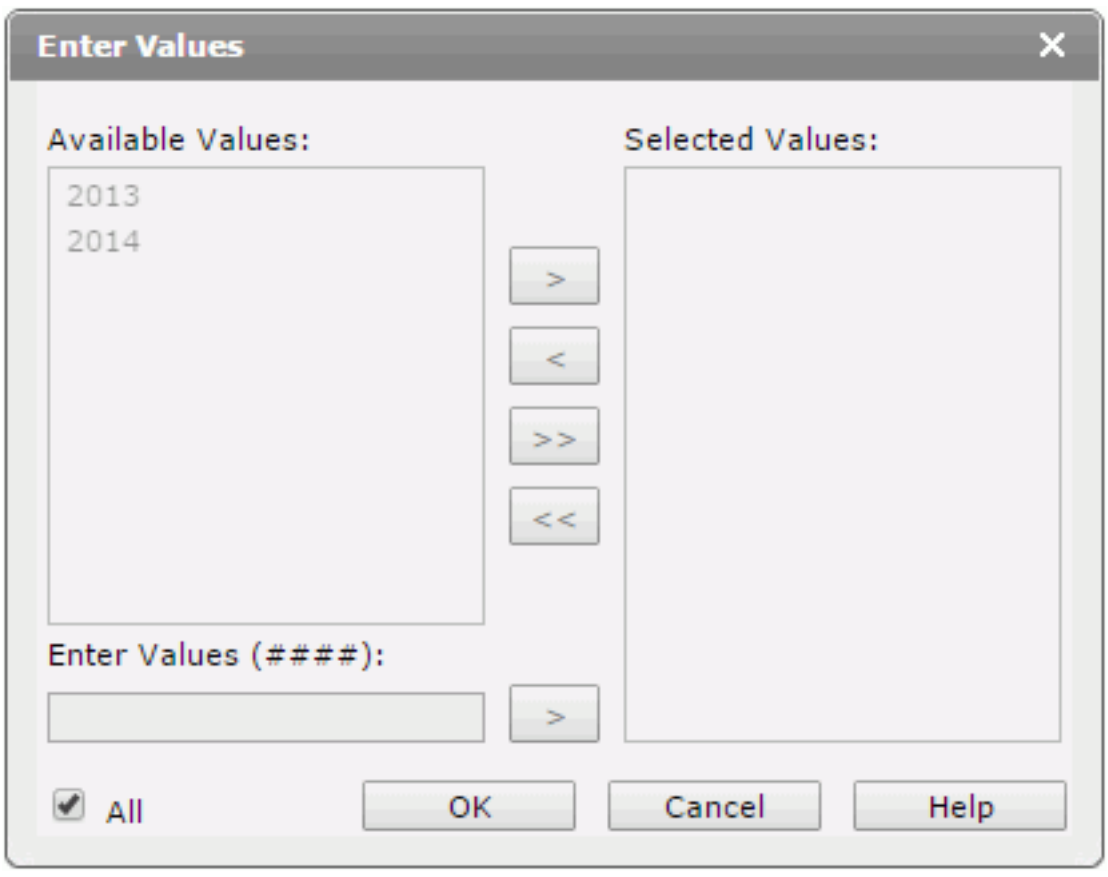
Displays the help document about this feature.

Related topics:

- Specifying parameter values

Enter Values dialog

The dialog appears when you click the button ... while specifying values for a parameter. It helps you to specify multiple values for the parameter.



Available Values

Lists all predefined parameter values for selection. When the parameter is bound with a column, but the Display Column is different from the Bind Column, values of the Display Column are listed here.

Selected Values

Lists the values selected. The selected values are case sensitive.

Adds the selected values from the Available Values box to the Selected Values box.

Removes the selected values from the Selected Values box.

Adds all the selected values from the Available Values box to the Selected Values box.

Removes all the selected values from the Selected Values box.

Enter Values

This option is available when the parameter's Allow Type-in of Value property is set to true in JReport Designer.

Enter a value manually in the text box and then click the button next to add the value to the Selected Values box. When the parameter is bound with a column, but the display column is different from the bind column, make sure the value you enter is that of the bind column.

Opens the Calendar dialog to specify a date and time value. Available only for Date, DateTime, or Time parameter which allows for type-in values and is not bound with any column.

Search

This option is available when the parameter's Allow Type-in of Value property is set to false in JReport Designer. It is used to search for values among the available values. After you type text in the Search text box, the matched text in the available values will be highlighted.

All

If selected, it means that the parameter value result is all the values in the DBMS. This option is available when the parameter's Enable the "All" Option property is set to true in JReport Designer. This is translated in SQL to remove the parameter which may select more values than listed in the available values list.

For example, when the parameter query is:

```
SELECT CUSTOMERS.CUSTOMERID,CUSTOMERS.CUSTOMERNAME FROM CUSTOMERS where CUSTOMERS.
CUSTOMERID>0 and CUSTOMERS.CUSTOMERID<4
```

When you click

to add 1, 2 and 3 as the parameter values, which are all the available values, the SQL is:

```
(CUSTOMERS.CUSTOMERID IN ( 1,2,3))
```

Run the same report again and this time check the All checkbox in the Enter Values dialog, the query is then:

```
( 1 = 1)
```

In this case, you will get more customers even though available values are only 1 •3.

Note: When a multi-value parameter is inserted as a field into a report and All is selected as the value, the field will show the string "All". If the parameter allows type-in values, "All" means all possible values of the parameter data type; if the parameter does not, "All" represents all the values of the parameter that come from the DBMS.

OK

Applies the selected values to the parameter and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Help

Displays the help document about this feature.

Folder Properties dialog

You can set the properties of a selected folder in its Properties dialog according to your requirements.

The Properties dialog is divided into the following two tabs:

- General
- Permission

Reset

Discards your modifications and restores the dialog to its default status.

OK

Retains the settings and submits the task to server.

Cancel

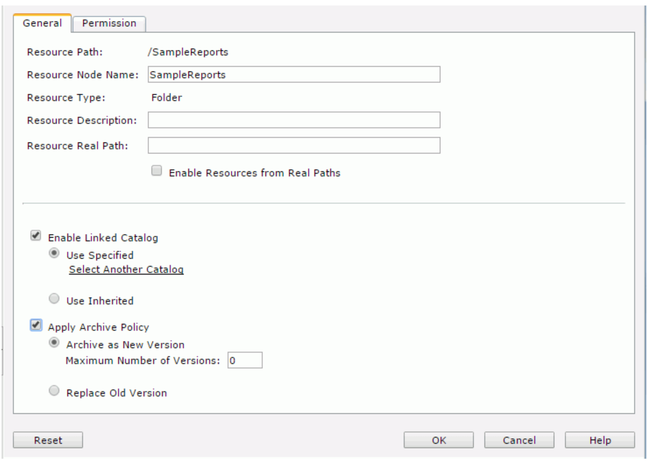
Cancels any settings and closes the dialog.

Help

Displays the help document about this feature.

General tab

Specifies the general properties of the folder.



Resource Path

Shows the resource path.

Resource Node Name

Specifies the name for the folder.

Resource Type

Shows the type of the resource.

Resource Description

Specifies description of the folder.

Resource Real Path

Specifies the real path of the folder.

Last Modified

Shows the last time the folder was modified. This option is available when the Properties dialog is accessed from the JReport Administration page.

Enable Resources from Real Paths

Specifies whether or not to enable getting resources from the folder's real path. Once enabled, the real path resources will be mapped into the resource node of the folder in the server resource tree and the server will always be able to get the resources and updates from the real path. For more information, refer to Getting and using resources from a real path.

Note: This option can only be checked when the Enable Resources from Real Paths option in the Configuration > Advanced panel on the JReport Administration page has also been checked.

[Custom Field Name]

Specifies value of the custom field for the folder. A custom field can be regarded as a resource property and is available when it is enabled.

Enable Linked Catalog

Enables to link the folder with a catalog.

If you have specified a linked catalog for the folder, then reports/library components and sub folders resided in the folder can inherit the linked catalog from the folder once their Enable Linked Catalog property is enabled.

If this option is unchecked, the reports/library components and sub folders resided in the folder cannot inherit linked catalog.

- **Use Specified**
  Links the folder with a catalog in the server resource tree.
  - **Select Another Catalog**
    Specifies another catalog in the Select Another Catalog dialog.
- **Use Inherited**
  Links the folder with the linked catalog inherited from its parent folder or from the server level if the folder is My Components, Public Components, My Reports, or Public Reports. Note that if the parent level does not enable linked catalog, you are not allowed to check this option.

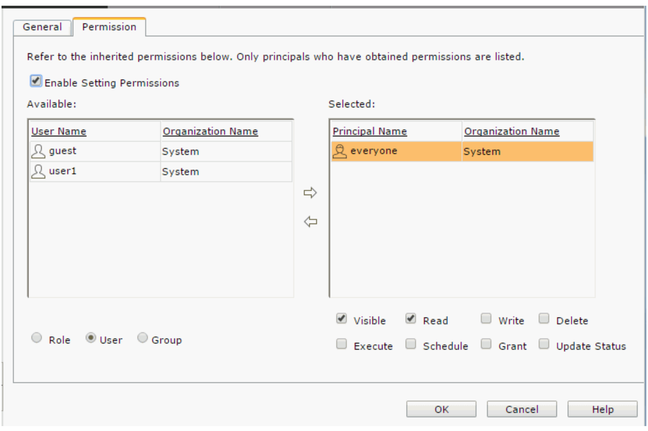
Apply Archive Policy

Applies an archive policy to resource versions in the folder.

- **Archive as New Version**
  Specifies to use multiple versions for the resources in the folder.
  - **Maximum Number of Versions**
    Specifies the maximum number of versions that will be listed in the version table of the resources. The default value is 0, which means that the version number is unlimited.
- **Replace Old Version**
  Specifies to replace the old version when a new version is generated.

Permission tab

Specifies permissions of roles/users/groups on the folder. This tab is available when the folder is in the Public Reports or Public Components folder.



Enable Setting Permissions

Enables the setting of permissions.

Available

Lists the roles/users/groups to which you can assign permissions. When organization is enabled, the Organization Name column will be available in the table.

- **Role**
  If checked, all roles will be displayed in the Available box for you to assign permissions.
- **User**
  If checked, all users will be displayed in the Available box for you to assign permissions.
- **Group**
  If checked, all groups will be displayed in the Available box for you to assign permissions.

Adds the selected role, user or group to the Selected box.

Removes the selected role, user or group from the Selected box.

Selected

Select a role/user/group in the Selected box and then select the permissions you would like the role/user/group to have on the folder. For what each permission is used for, see Permission.

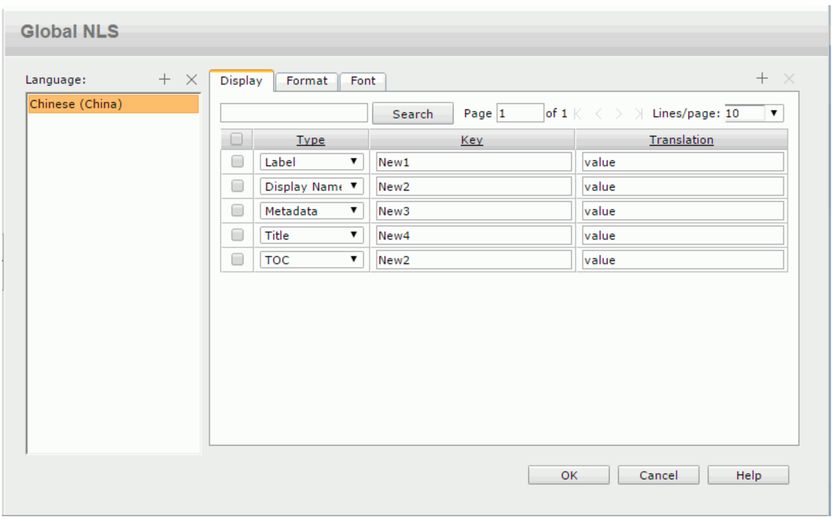
When organization is enabled, the Organization Name column will be available in the table.

Related topics:

- Changing resource and folder properties

Global NLS dialog

This dialog appears when you click Resources > Global NLS on the JReport Administration page.It helps you to create global NLS resources which can be used globally for all reports, library components and Visual Analysis templates in both JReport Server and JReport Designer.



Language

Specifies the languages into which the display text will be translated.

Opens the Select Language Source dialog to choose where to add a language.

Removes the selected language from the Language box.

Display tab

Specifies the translation version of the display text in the target language.

- **Search**
  Searches for the Key and Translation items that match the search condition in the Search text box. Here you are not allowed to search wildcards. The results will be paged and listed in the table below.
- Page 3 of 4
  Displays the current page number. You can also input a page number in the page box and press Enter on the keyboard to go to that page.
-
  Goes to the first page.
-
  Goes to the previous page.
-
  Goes to the next page.
-
  Goes to the last page.
- Lines/page: 10
  Specifies how many lines will be displayed in each page.
- **Checkbox**
  Specifies to select the display item. Check the checkbox on the column header to select all the items.
- **Type**
  Specifies the type of display text for different objects.
  - **Column**
    This type is only for page reports running in Page Report Studio. It is the type of display text of columns.
  - **DisplayName**
    Type of display text of object display name.
  - **Metadata**
    Type of display text of metadata. Metadata mainly refers to catalog resources, such as table/view columns, business views, formulas, summaries, parameters, and so on.
  - **Label**
    Type of display text of label, some web controls and UDOs.
  - **Prompt**
    Type of display text of parameter prompt value.
  - **Title**
    Type of display text of filter control, library component and objects in it.
  - **TOC**
    Type of display text on the TOC tree.
- **Key**
  Lists keys to indicate the objects in the original language.
- **Translation**
  Specifies to which the display text will be translated in the target language.
-
  Specifies to add a new row of display for the target language.
-
  Specifies to remove the selected rows of display for the target language.

Format tab

Specifies format properties for the target language.

- **Checkbox**
  Specifies to select the format item. Check the checkbox on the column header to select all the items.
- **Key**
  Specifies keys to indicate the formats in the original language.
- **Format**
  Specifies the formats for the keys in the target language.
-
  Specifies to add a new row of format for the target language.
-
  Specifies to remove the selected rows of format for the target language.

Font tab

Specifies font properties for the target language.

- **Checkbox**
  Specifies to select the font item. Check the checkbox on the column header to select all the items.
- **Key**
  Specifies keys to indicate the fonts in the original language. Choose from the drop-down lists the font face and font size for the key.
- **Font Face**
  Specifies the font face for the key in the target language.
- **Font Size**
  Specifies the font size for the key in the target language.
  - **Use Relative Font Size**
    Specifies to use a relative font size, which means the font size can be adjusted according to the font size settings in the web browser.
-
  Specifies to add a new row of font for the target language.
-
  Specifies to remove the selected rows of font for the target language.

OK

Applies the settings and closes the dialog.

Cancel

Cancels the settings and exits the dialog.

Help

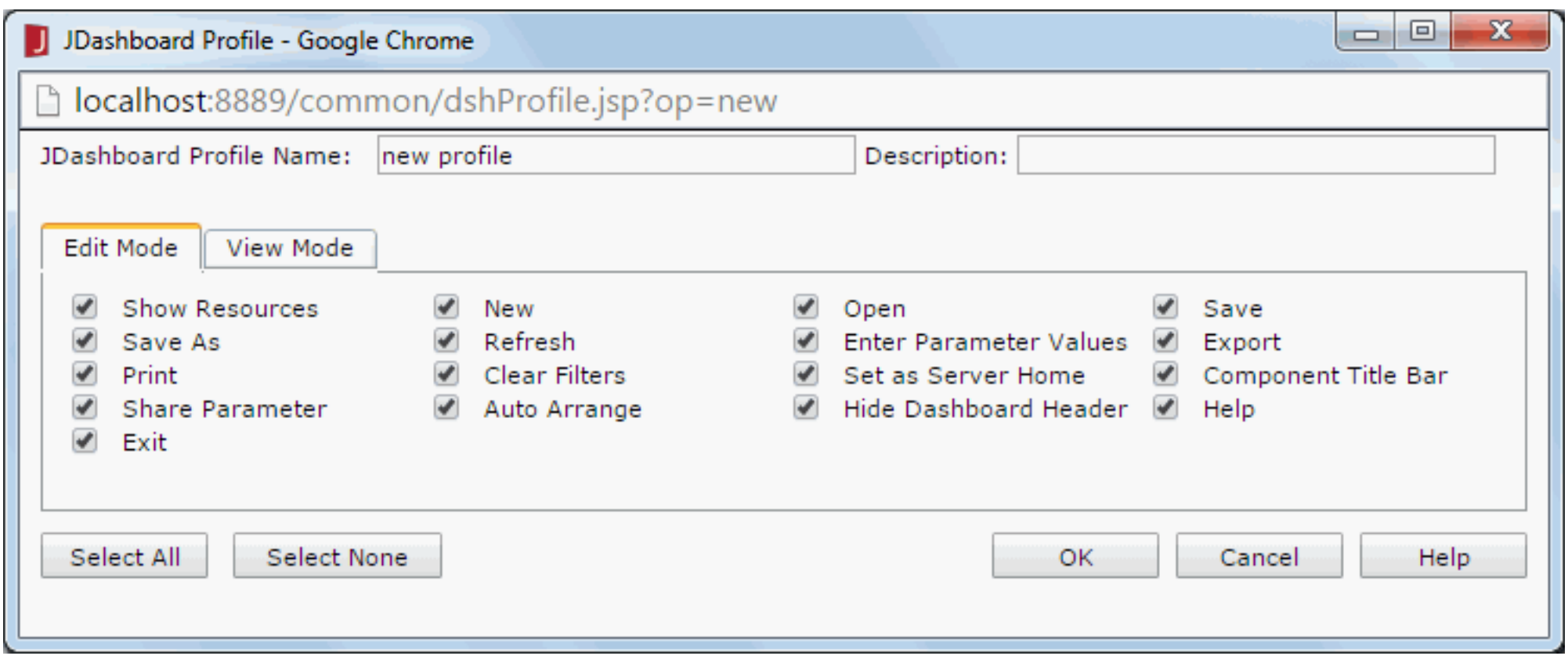
Displays the help document about this feature.

Related topics:

- Creating global NLS

JDashboard Profile dialog

This dialog appears when you click the New Profile link in the Profile > Customize Profile > JDashboard > Features tab on the JReport Administration page. It helps you to create profiles for applying in JDashboard.



JDashboard Profile Name

Specifies the name for the new profile.

Description

Specifies information about the new profile.

Edit Mode

In this tab, you can turn on/off the required JDashboard edit mode features for the profile.

- **Show Resource**
  Shows the button which is the entry to the Resources panel that provides library components, report components, and some other objects for adding into dashboards.
- **New**
  Creates a new dashboard.
- **Open**
  Specifies a dashboard to open.
- **Save**
  Saves the changes made to the current dashboard.
- **Save As**
  Saves the current dashboard with a different name or to a new location.
- **Refresh**
  Refreshes the current dashboard.
- **Enter Parameter Value**
  Lists all the parameters used in the dashboard for specifying their values.
- **Export**
  Exports the components in the current dashboard.
- **Print**
  Prints the components in the current dashboard.
- **Clear Filter**
  Removes all the filters from the current dashboard including those generated via sliders, filter controls, messages, drilling and going actions, and those designed using web browsers such as Page Report Studio and Web Report Studio, except query filters and others designed and taken effect in JReport Designer.
- **Set as Server Home**
  Sets the current JDashboard status as the home page after logging onto JReport Server.
- **Component Title Bar**
  Customizes the way of showing component title bar and the icons on it.
- **Theme**
  Specifies a theme to apply to the current dashboard.
- **Share Parameter**
  Shares parameters between library components.
- **Auto Arrange**
  Automatically arranges the library components in the current dashboard to be tidy. When this option is unchecked, both Arrange and Auto Arrange options are disabled on JDashboard window.
- **Hide JDashboard Header**
  Changes the current status of the dashboard header from being shown to hidden or from being hidden to shown.
- **Help**
  Displays the JDashboard help documents.
- **Exit**
  Exits JDashboard.

View Mode

In this tab, you can turn on/off the required JDashboard view mode features for the profile.

- **Refresh**
  Refreshes the current dashboard.
- **Enter Parameter Value**
  Lists all the parameters used in the dashboard for specifying their values.
- **Export**
  Exports the components in the current dashboard.
- **Print**
  Prints the components in the current dashboard.
- **Clear Filter**
  Removes all the filters from the current dashboard including those generated via sliders, filter controls, messages, drilling and going actions, and those designed using web browsers such as Page Report Studio and Web Report Studio, except query filters and others designed and taken effect in JReport Designer.
- **Set as Server Home**
  Sets the current JDashboard status as the home page after logging onto JReport Server.
- **Component Title Bar**
  Customizes the way of showing component title bar and the icons on it.
- **Theme**
  Specifies a theme to apply to the current dashboard.
- **Share Parameter**
  Shares parameters between library components.
- **Hide JDashboard Header**
  Changes the current status of the dashboard header from being shown to hidden or from being hidden to shown.
- **Help**
  Displays the JDashboard help documents.
- **Exit**
  Exits JDashboard.

Select All

Has all options in the tabs checked.

Select None

Has all options in the tabs unchecked.

OK

Finishes creating the profile and leaves the dialog.

Cancel

Cancels creating the profile and leaves the dialog.

Help

Displays the help document about this feature.

Library Component Properties dialog

You can set the properties of a library component in its Properties dialog according to your requirements.

The Properties dialog is divided into the following two tabs:

- General
- Permission

Reset

Discards your modifications and restores the dialog to its default status.

OK

Closes this dialog and applies the properties to the library component.

Cancel

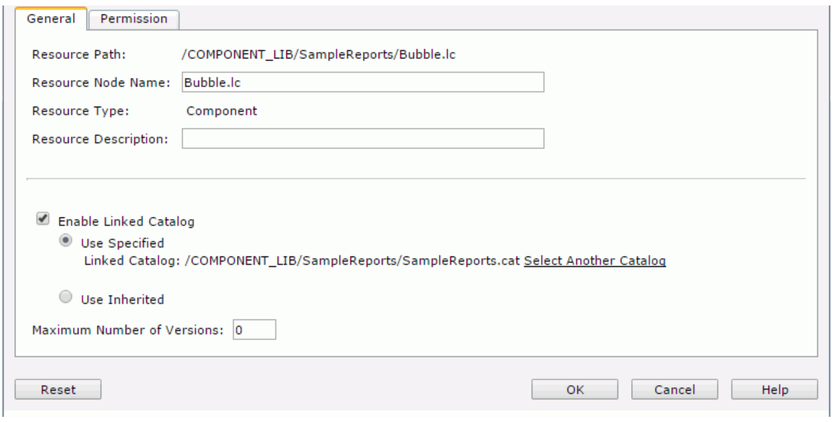
Cancels any settings and closes the dialog.

Help

Displays the help document about this feature.

General tab

Specifies the general properties of the library component.



Resource Path

Shows the resource path.

Resource Node Name

Specifies the name for the library component.

Resource Type

Shows the type of the resource.

Resource Description

Specifies description of the library component.

[Custom Field Name]

Specifies value of the custom field for the library component. A custom field can be regarded as a resource property and is available when it is enabled.

Enable Linked Catalog

Enables to link the library component with a catalog. If you have linked a library component with a catalog, then even if the library component and the catalog are not in the same directory, it can also run with the catalog.

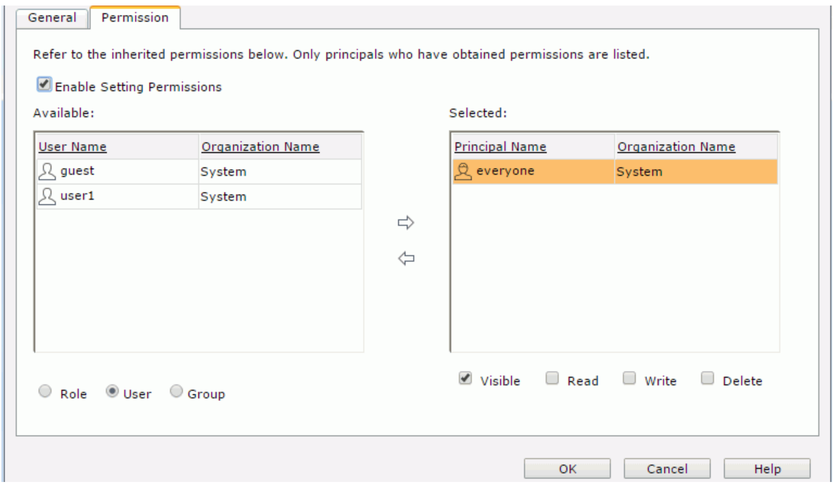
- **Use Specified**
  Links the library component with a catalog in the server resource tree.
  - **Select Another Catalog**
    Specifies another catalog in the Select Another Catalog dialog.
- **Use Inherited**
  Links the library component with the linked catalog inherited from its parent folder. Note that if the parent level does not enable linked catalog, you are not allowed to check this option.

Maximum Number of Versions

Specifies the maximum version amount that will be listed in the version table. By default the version amount is unlimited.

Permission tab

Specifies permissions of roles/users/groups on the library component. This tab is available when the library component is in the Public Components folder.



Enable Setting Permissions

Enables the setting of permissions.

Available

Lists the roles/users/groups to which you can assign permissions. When organization is enabled, the Organization Name column will be available in the table.

- **Role**
  If checked, all roles will be displayed in the Available box for you to assign permissions.
- **User**
  If checked, all users will be displayed in the Available box for you to assign permissions.
- **Group**
  If checked, all groups will be displayed in the Available box for you to assign permissions.

Adds the selected role, user or group to the Selected box.

Removes the selected role, user or group from the Selected box.

Selected

Select a role/user/group in the Selected box and then select the permissions you would like the role/user/group to have on the library component. For what each permission is used for, see Permission.

When organization is enabled, the Organization Name column will be available in the table.

Related topics:

- Changing resource and folder properties

New Cache dialog for scheduling

The dialog is displayed when you click the OK button in the New Cache dialog for selecting queries. It helps you to define the updating policy of the data caches created based on the selected queries.

This dialog contains the tabs listed below. Set the settings in each tab according to your requirements.

- General tab
- Conditions tab
- Notification tab

Back
Goes back to the previous tab.

Next
Goes to the next tab.

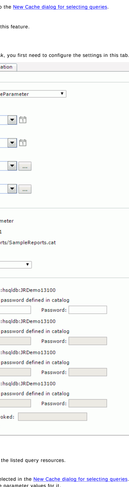
Finish
Applies the settings.

Cancel
Cancels any settings and returns to the New Cache dialog for selecting queries.

Help
Displays the help document about this feature.

General tab

Before you can schedule a CRD task, you first need to configure the settings in this tab.



Enter Parameters section

Specifies the parameter values for the listed query resources.

Select Query
Lists all the query resources you selected in the New Cache dialog for selecting queries. Select a query resource from the drop-down list and then specify the parameter values for it.

Enter Parameters
Specifies values of the parameters applied in the selected query resource. If the query resource does not have parameters, "No Parameter Needed" will be displayed here.

You may specify parameter values in these ways.

Cached Report Data Info section

Specifies the data cache information.

Name
Displays the name of the selected query resource. It is also used as the name of the new data cache.

Data Source
Displays the data source name.

Catalog
Displays the catalog information. The latest version is used by default.

Type
Displays the type of the query resource.

Priority
Specifies a priority level to the scheduling task. The priority levels are from 1 to 10 in ascending order of lowest priority to highest priority. By default this property is ignored unless server properties is modified to set queue.policy not equal to 0.

Advanced section

Configures some advanced settings.

Connect to [Data Source Name]
Specifies the DB user and password with which you want to connect to the data source.

- **Use the DB user and password defined in catalog**
  If checked, the DB user and password defined in the catalog will be used.
- **Use the DB User**
  If checked, specify another DB user and password instead of the one defined in the catalog.

Add TaskListener to be Invoked
Enables you to call the Java application before/after viewing or when scheduling the task.

Enable Auto Recover Task
Specifies to enable the task to be auto recovered.

- **Maximum Retry Times**
  Specifies the maximum number of times in which to retry running the task in order to recover it.
- **Retry Interval**
  Specifies the interval between retries.
- **Recreate All Results**
  Specifies whether to recreate all or just the failed results.

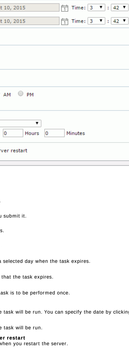
Conditions tab

The Conditions tab allows you to specify the conditions for the task. It contains the following two sub tabs:

- Time tab
- Trigger tab

Time tab

The Time tab allows you to set settings for specifying the time for when a task is to be performed.



Time Zone
Specifies the time zone.

Time Type
Specifies the time type of the task.

- **Run this task immediately**
  Performs the task as soon as you submit it.
- **Expires**
  Specifies when the task expires.
  - **Never**
    The new task never expires.
  - **At**
    Specifies a specific time on a selected day when the task expires.
  - **After**
    Specifies a time period after that the task expires.
- **Run this task at**
  Specifies the time for when the task is to be performed once.
  - **Date**
    Specifies the date for when the task will be run. You can specify the date by clicking the calendar button.
  - **Time**
    Specifies the time for when the task will be run.
- **Run missed task upon Server restart**
  Specifies to run missed tasks when you restart the server.
- **Run this task periodically**
  Specifies the time for when the task is to be performed on a repeated basis.
  - **Do not start schedule until the report is requested**
    This option is used to freeze the way of automatically performing the schedule and change it to the schedule being fired upon certain condition.
    If the option is not checked, the schedule will be performed automatically according to the time condition.
    If the option is checked, the schedule will be fired only upon the first report running request based on the query after each scheduled time.
  - **Duration**
    - **Run After**
      Specifies the start date and time of the period during which the task is to be performed on a repeated basis.
    - **Run until**
      Specifies the end date and time of the period during which the task is to be performed on a repeated basis.
  - **Date**
    Specifies the date for when a task is to be performed.
    - **Daily**
      Performs the task every one day, two days, etc., or every weekday (from Monday to Friday).
    - **Weekly**
      Performs the task every one week, two weeks, etc., and on Sunday, Monday, Tuesday, etc.
    - **Monthly**
      Performs the task on the first day, second day, etc. of every one month, two months, etc., or on the first Sunday, Monday, etc. of every one month, two months, etc.
  - **Time**
    Specifies the exact time for when a task is to be performed on a selected day.
    - **At**
      Specifies a specific time for when to perform a task on a selected day.
    - **Hourly**
      Performs the task every one hour, two hours, etc., at a certain minute on a selected day.
    - **Minutely**
      Performs the task every one minute, two minutes, etc.
  - **Expires**
    Specifies the expiration time for how long to keep the scheduled data caches.
    - **Never**
      The data caches will not expire.
    - **At**
      Specifies a specific time when the data caches will expire.
    - **After**
      Specifies a period of time after which the data caches will expire.
- **Run missed task upon server restart**
  Specifies to run missed tasks when you restart the server.

Trigger tab

The Trigger tab allows you to set settings for specifying a trigger for the task.

Select a trigger to bind
Specifies the trigger from the drop-down list for the task.

Create New
Creates a new trigger.

- **Trigger Name**
  Specifies the name of the trigger.
- **Conflict Check**
  Checks if the name has been used for another trigger.
- **Description**
  Specifies the description of the trigger (if required).
- **OK**
  Creates the trigger with the specified information.
- **Cancel**
  Cancels the trigger creating process.

Logic with time condition
Specifies the logic between time condition and trigger condition.

- **Trigger Only**
  Performs the task only when the trigger fires.
- **Trigger and Time Condition**
  Performs the task when both time condition and trigger condition are ready.
  Notes:
  - No matter which condition is ready, the task can only be performed when its counterpart is ready.
  - If you specify the task to be performed at a specific time, you must check the checkbox Run missed task upon Server restart, otherwise the task will be regarded as expired and will be deleted when the time condition is ready before the trigger condition.
- **Time Condition after Trigger**
  Performs a task when both the time condition and the trigger condition are ready. The trigger condition must be ready before the time condition. If the time condition is ready before the trigger condition, the task will be regarded as expired and will be deleted.
- **Time Condition or Trigger**
  Performs the task when either the time condition or the trigger condition is ready.

Notification tab

The Notification tab allows you to notify someone by e-mail when a task has finished running, regardless of whether it was successful or unsuccessful.

When task is successful
Specifies to send an e-mail when the task is successful.

When task fails
Specifies to send an e-mail when the task is unsuccessful.

To
Specifies the address you want to send the e-mail to.

Cc
Specifies the address you want to copy to.

Bcc
Specifies the address you want to secretly copy to.

Subject
Specifies the subject of the e-mail.

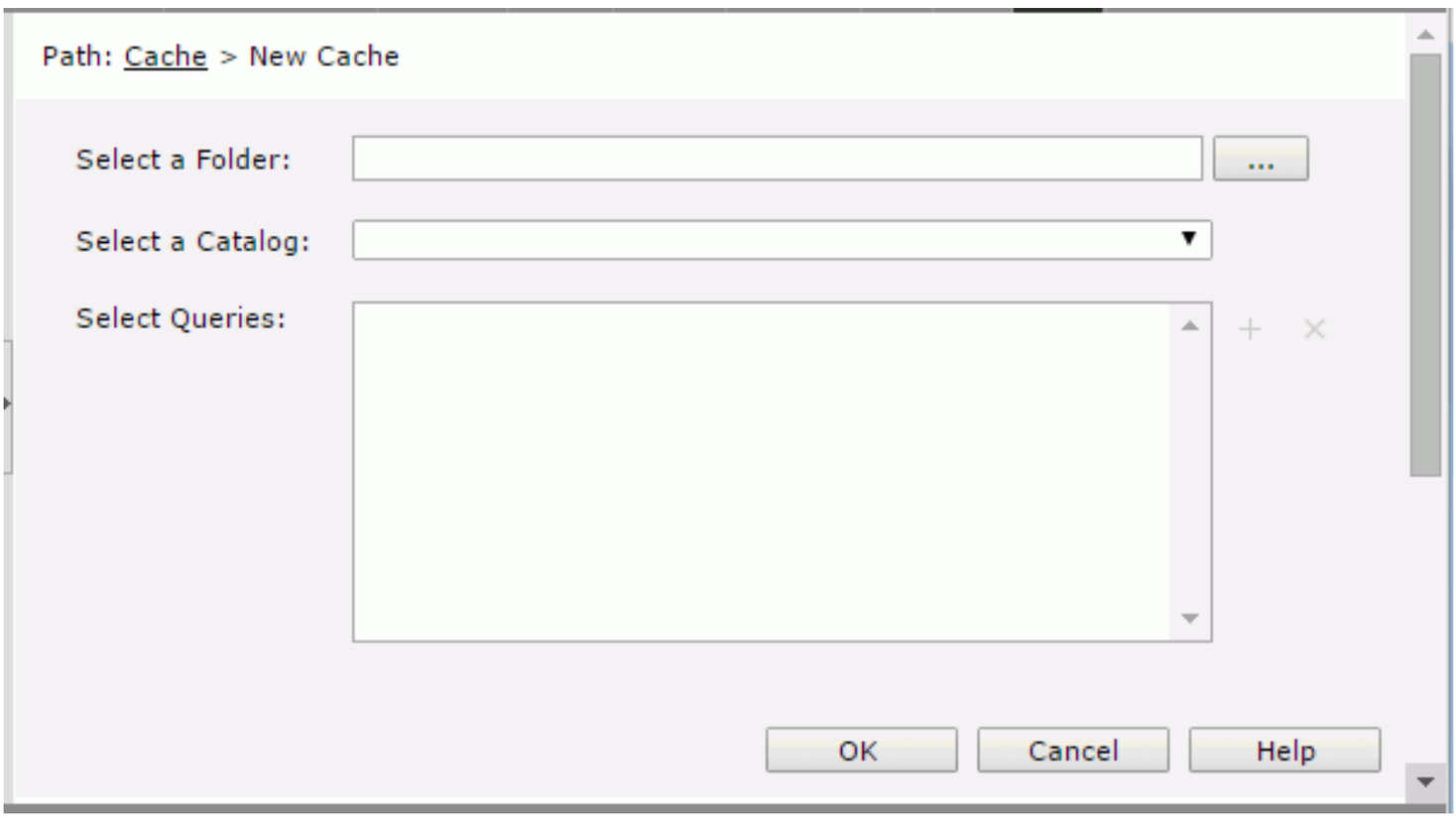
Comments
Specifies the contents of the mail or comments to the contents.

Related topics:

- Managing cached report data

New Cache dialog for selecting queries

The dialog appears when you click the New Cache link on the JReport Administration page > Cache > Data Cache > Cached Report Data tab. It helps you to select queries which you will create data caches for.



Select a Folder

Specifies the folder in the server resource tree that contains the required catalog. You can click the button ... to open the Select Folder dialog for selecting the folder.

Select a Catalog

Specifies the catalog in the folder.

Select Queries

Lists the selected queries.

- +
  Opens the Select Queries dialog to select the queries that you are going to create data caches for.
- ×
  Removes the selected queries.

OK

Redirects to the New Cache dialog for scheduling.

Cancel

Does not retain any changes and closes the dialog.

Help

Displays the help document about this feature.

Related topics:

- Managing cached report data

New Cube dialog for scheduling

The dialog is displayed when you click the OK button in the New Cube dialog for selecting business view. It helps you to define the updating policy of the in-memory cube created based on the selected business view.

This dialog contains the tabs listed below. Set the settings in each tab according to your requirements.

- General tab
- Schedule tab
- Settings tab
- Notification tab

Back
Goes back to the previous tab.

Next
Goes to the next tab.

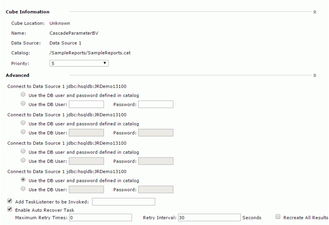
Finish
Applies the settings.

Cancel
Cancels any settings and returns to the New Cube dialog for selecting business view.

Help
Displays the help document about this feature.

General tab

Before you can schedule a cube task, you first need to configure the settings in this tab.



Enter Parameters

Specifies values of the parameters applied in the selected business view. If the business view does not have parameters, "No Parameter Needed" will be displayed here.

You may specify parameter values in three ways.

Cube Information

Specifies the in-memory cube information.

Cube Location
Shows the location of the cube. It is "Unknown" if it is the first time you schedule a task for the business view.

Name
Shows the name of the business view. It is also used as the name of the new cube.

Data Source
Shows the data source name.

Catalog
Shows the catalog name.

Priority
Specifies a priority level to the scheduled task. The priority levels are from 1 to 10 in ascending order of lowest priority to highest priority. By default this property is ignored unless server properties is modified to set queue policy not equal to 0.

Advanced

Configures some advanced settings.

Connect to [Data Source Name]
Specifies the DB user and password with which you want to connect to the data source.

- **Use the DB user and password defined in catalog**
  If checked, the DB user and password defined in the catalog will be used.
- **Use the DB User**
  If checked, specify another DB user and password instead of the one defined in the catalog.

Add TaskListener to be Invoked
Enables you to call the Java application before/after viewing or when scheduling the task.

Enable Auto Recover Task
Specifies to enable the task to be auto recovered.

- **Maximum Retry Times**
  Specifies the maximum number of times in which to retry running the task in order to recover it.
- **Retry Interval**
  Specifies the interval between retries.
- **Recreate All Results**
  Specifies whether to recreate all results or just the failed results.

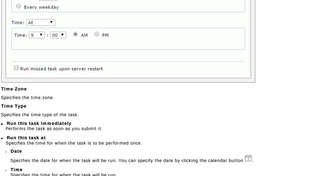
Schedule tab

The Conditions tab allows you to specify the conditions for the task. It contains the following two sub tabs:

- Time tab
- Trigger tab

Time tab

The Time tab allows you to set settings for specifying the time for when a task is to be performed.



Time Zone
Specifies the time zone.

Time Type
Specifies the time type of the task.

- **Run this task immediately**
  Performs the task as soon as you submit it.
- **Run this task at**
  Specifies the time for when the task is to be performed once.
  - **Date**
    Specifies the date for when the task will be run. You can specify the date by clicking the calendar button.
  - **Time**
    Specifies the time for when the task will be run.
- **Run missed task upon Server restart**
  Specifies to run missed tasks when you restart the server.
- **Run this task periodically**
  Specifies the time for when the task is to be performed on a repeated basis.
  - **Do not start schedule until the report is requested**
    This option is used to freeze the way of automatically performing the schedule and change it to the schedule being fired upon certain condition.
    If the option is not checked, the schedule will be performed automatically according to the time condition.
    If the option is checked, the schedule will be fired only upon the first report running request based on the business view after each scheduled time.
  - **Duration**
    - **Run after**
      Specifies the start date and time of the period during which the task is to be performed on a repeated basis.
    - **Run until**
      Specifies the end date and time of the period during which the task is to be performed on a repeated basis.
  - **Date**
    Specifies the date for when a task is to be performed.
    - **Daily**
      Performs the task every one day, two days, etc., or every weekday (from Monday to Friday).
    - **Weekly**
      Performs the task every one week, two weeks, etc., and on Sunday, Monday, Tuesday, etc.
    - **Monthly**
      Performs the task on the first day, second day, etc. of every one month, two months, etc., or on the first Sunday, Monday, etc. of every one month, two months, etc.
  - **Time**
    Specifies the exact time for when a task is to be performed on a selected day.
    - **At**
      Specifies a specific time for when to perform a task on a selected day.
    - **Hourly**
      Performs the task every one hour, two hours, etc., at a certain minute on a selected day.
    - **Minutely**
      Performs the task every one minute, two minutes, etc.
- **Run missed task upon server restart**
  Specifies to run missed tasks when you restart the server.

Trigger tab

The Trigger tab allows you to set settings for specifying a trigger for the task.

Select a trigger to bind
Specifies the trigger from the drop-down list for the task.

Create New
Creates a new trigger.

- **Trigger Name**
  Specifies the name of the trigger.
- **Conflict Check**
  Checks if the name has been used for another trigger.
- **Description**
  Specifies the description of the trigger (if required).
- **OK**
  Creates the trigger with the specified information.
- **Cancel**
  Cancels the trigger creating process.

Logic with time condition
Specifies the logic between time condition and trigger condition.

- **Trigger Only**
  Performs the task only when the trigger fires.
- **Trigger and Time Condition**
  Performs the task when both time condition and trigger condition are ready.
  - **Notes:**
    - No matter which condition is ready, the task can only be performed when its counterpart is ready.
    - If you specify the task to be performed at a specific time, you must check the checkbox Run missed task upon Server restart, otherwise the task will be regarded as expired and will be deleted when the time condition is ready before the trigger condition.
- **Time Condition after Trigger**
  Performs the task when both time condition and the trigger condition are ready. The trigger condition must be ready before the time condition. If the time condition is ready before the trigger condition, the task will be regarded as expired and will be deleted.
- **Time Condition or Trigger**
  Performs the task when either the time condition or the trigger condition is ready.

Settings tab

The Settings tab allows you to configure the location and size of the cube.

Maximum Memory Allowed
Shows the maximum memory for the cube to use. Each cube has this setting. The total of all cubes' maximum memory allowed should be no more than the maximum cube memory allowed.
The value must be provided, otherwise you are not allowed to continue with the other tabs or to finish the dialog. The value should be no more than the available memory.

Move the cube to disk if insufficient memory was allocated
When the option is selected, the cube will be cached on disk when the specified memory size is not enough to hold it.
If the option is selected, when initiating or updating an in-memory cube, if the memory requirement is bigger than the cube's maximum memory allowed or the available memory cubes' memory usage will be bigger than the maximum cube memory allowed, the cube will be swapped to disk. Otherwise, if this option is unchecked, the cube status will be set to disabled.

Cache Detail on Disk
When the option is selected, the aggregation data in the business view will be cached into memory and the detail data will be cached on disk.
When the option is unselected, only the aggregation data in the business view will be cached into memory.

Available Memory
Shows the memory that can be used to run the cube. The value equals the maximum cube memory allowed for all cubes minus the total memory usage.

Total Memory Usage
Shows the total maximum memory allowed for all in-memory cubes which are not in the Disabled status.

Memory usage table
Lists the memory usage of all the other cubes excluding the current.

- **Name**
  The name of the cube.
- **Maximum Memory Usage**
  The maximum memory allowed for the cube.

Notification tab

The Notification tab allows you to notify someone by e-mail when a task has finished running, regardless of whether it was successful or unsuccessful.

When task is successful
Specifies to send an e-mail when the task is successful.

When task fails
Specifies to send an e-mail when the task is unsuccessful.

To
Specifies the address you want to send the e-mail to.

Cc
Specifies the address you want to copy to.

Bcc
Specifies the address you want to secretly copy to.

Subject
Specifies the subject of the e-mail.

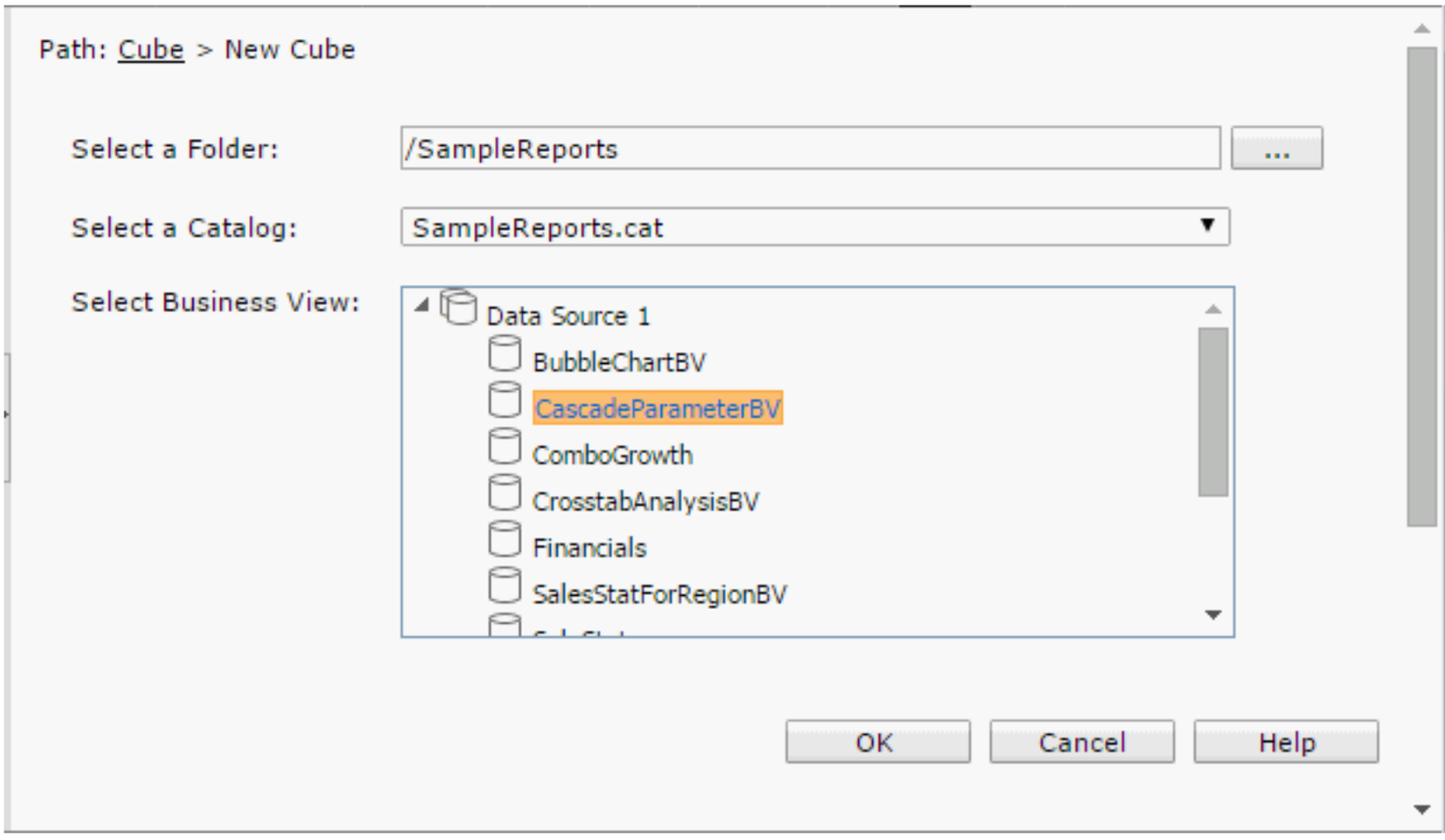
Comments
Specifies the contents of the mail or comments to the contents.

Related topics:

- Creating in-memory cubes

New Cube dialog for selecting business view

The dialog appears when you click the New Cube link on the JReport Administration page > Cube panel. It allows you to select a business view which you will create in-memory cube for.



Select a Folder

Specifies the folder in the server resource tree that contains the required catalog. You can click the button ... to open the Select Folder dialog for selecting the folder.

Select a Catalog

Specifies the catalog in the folder.

Select Business View

Specifies a business view in the catalog.

OK

Redirects to the New Cube dialog for scheduling.

Cancel

Does not retain any changes and closes the dialog.

Help

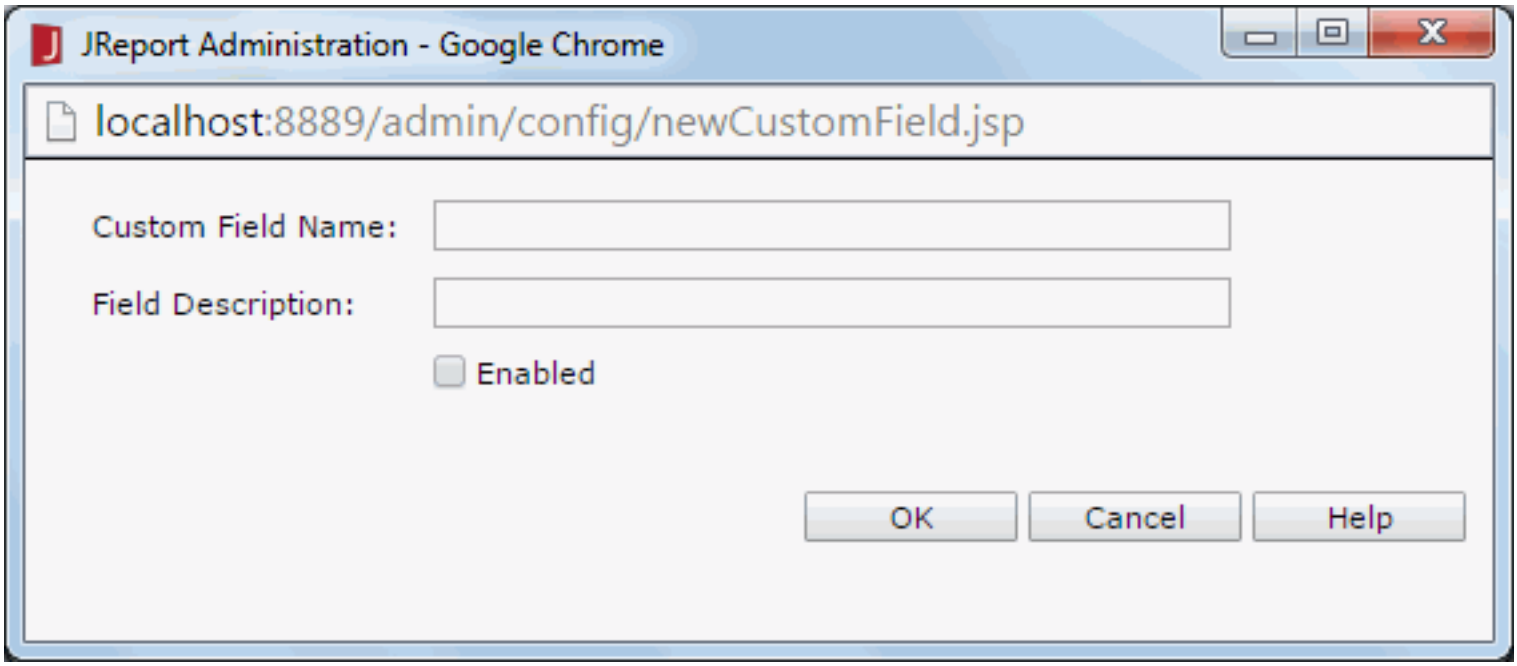
Displays the help document about this feature.

Related topics:

- Creating in-memory cubes

New Custom Field dialog

This dialog appears when you click the New Custom Field link in the Configuration > Custom Field panel on the JReport Administration page. It is used to create user defined fields which can be used as resource properties.



Custom Field Name

Specifies the name for the custom field.

Field Description

Specifies a description for the custom field.

Enabled

Specifies whether to enable the custom field. The enabled custom fields are available for showing in the resource properties.

OK

Creates the custom field and exits the dialog.

Cancel

Cancels operations and closes the dialog.

Help

Displays the help document about this feature.

Related topics:

- Working with custom fields

New Dashboard Listener dialog

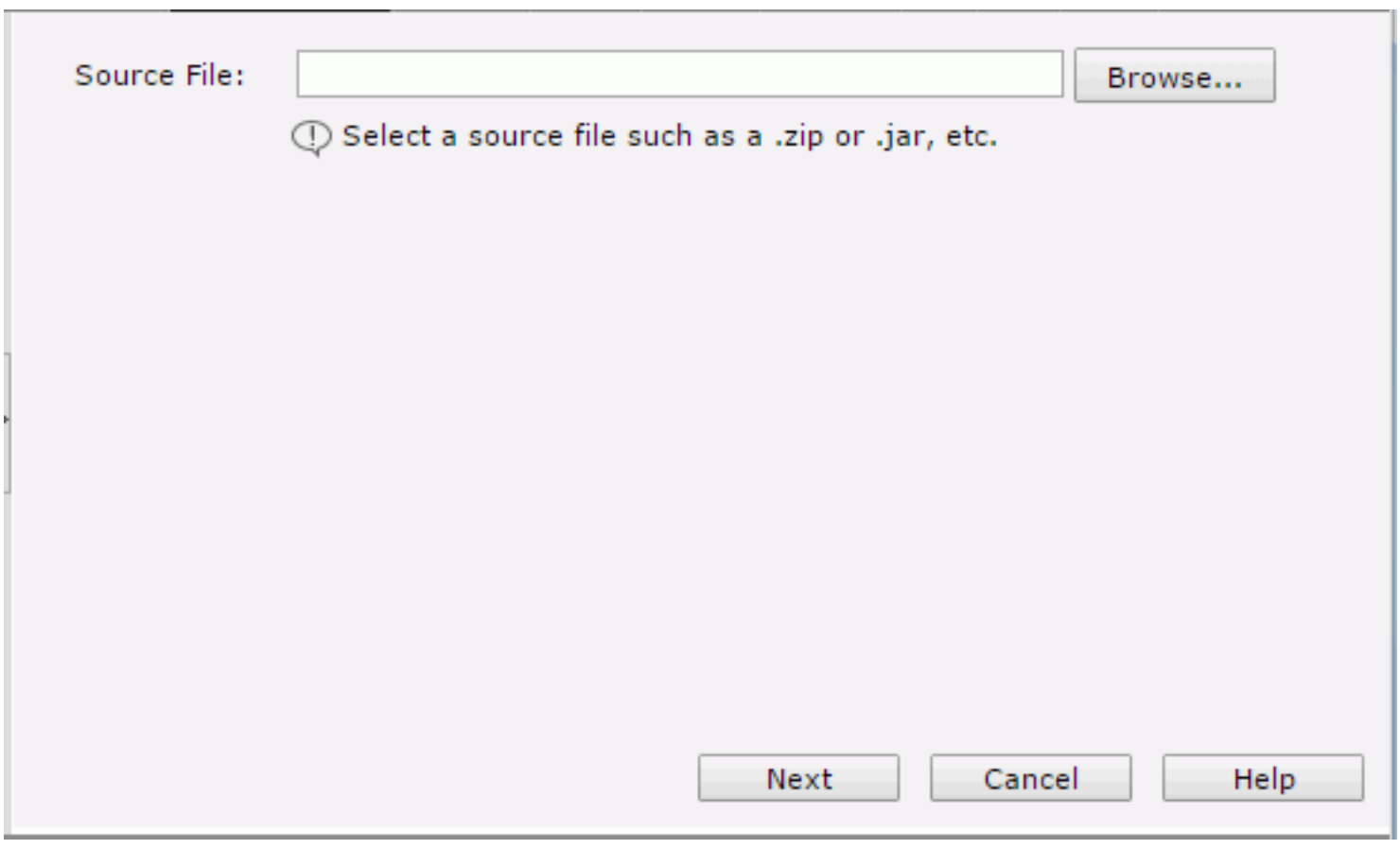
This dialog is used for uploading and registering an implementation of the Dashboard Listener API.

It is divided into two steps:

- Step 1: Specifying an implementation file
- Step 2: Specifying target resources

Step 1: Specifying an implementation file

This is used for adding an implementation file of the Dashboard Listener API.



Source File

Uploads an implementation class file of the Dashboard Listener API which could be a .zip or .jar file using the Browse button.

Next

Moves forward to the next step of specifying the target resources.

Cancel

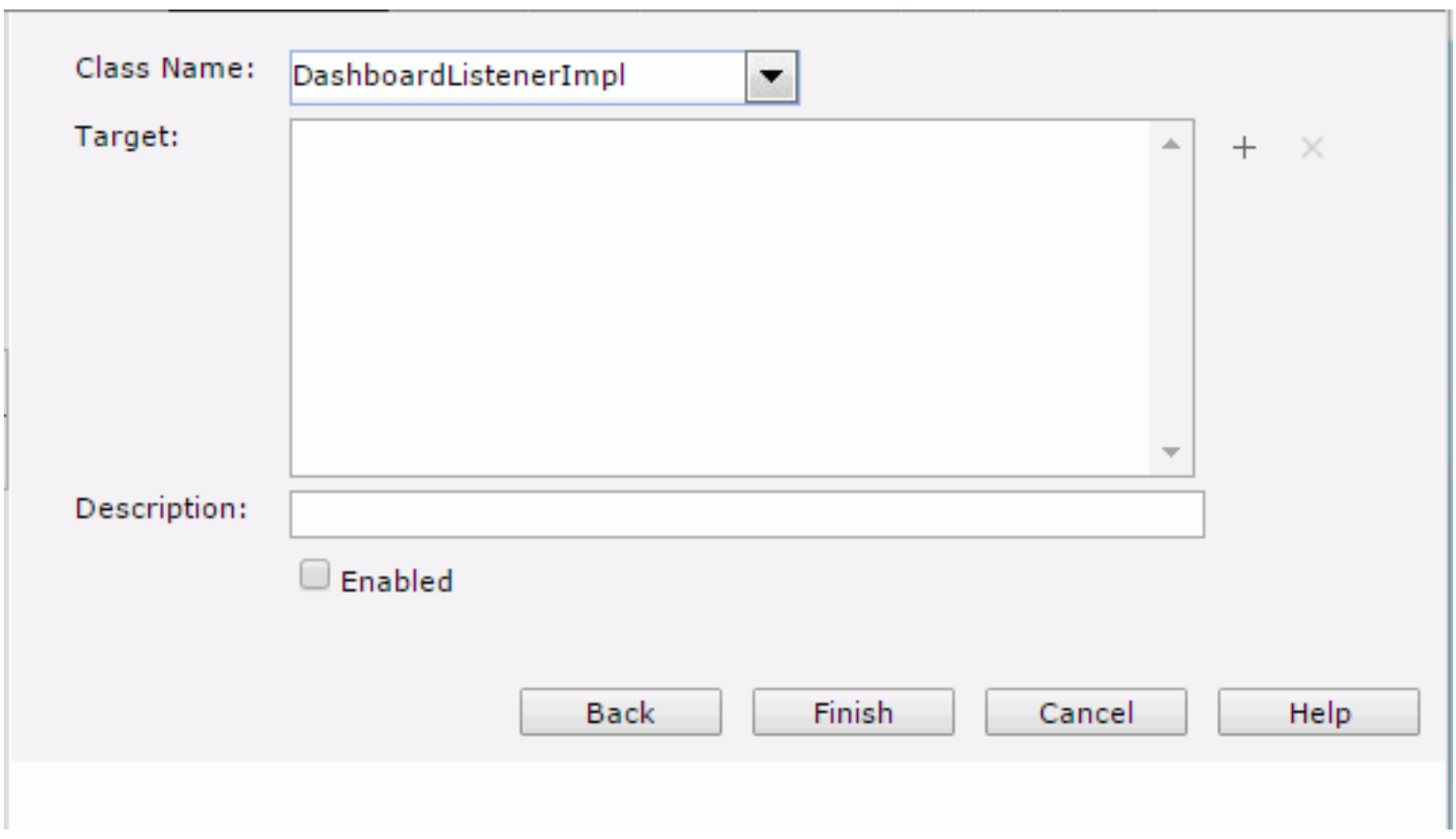
Cancels the setting and closes the dialog.

Help

Displays the help document about this feature.

Step 2: Specifying target resources

This is used for specifying the target resources that the implementation will take effect on.



Class Name

Lists class names defined in the specified implementation file. Select a class name from the drop-down list or input the name manually.

Target

Specifies the resources that the implementation will take effect on. Multiple selection is supported in the Target box.

If the target value is null, the implementation will not take effect.

-
  Opens the Select Target dialog for adding a new target.
-
  Removes the selected target values.

Description

Specifies the description of the implementation.

Enabled

Specifies whether the implementation is enabled. The enabled implementations will take effect.

Back

Returns to the step of specifying an implementation file.

Finish

Adds the implementation and exits the dialog.

Cancel

Cancels the setting and closes the dialog.

Help

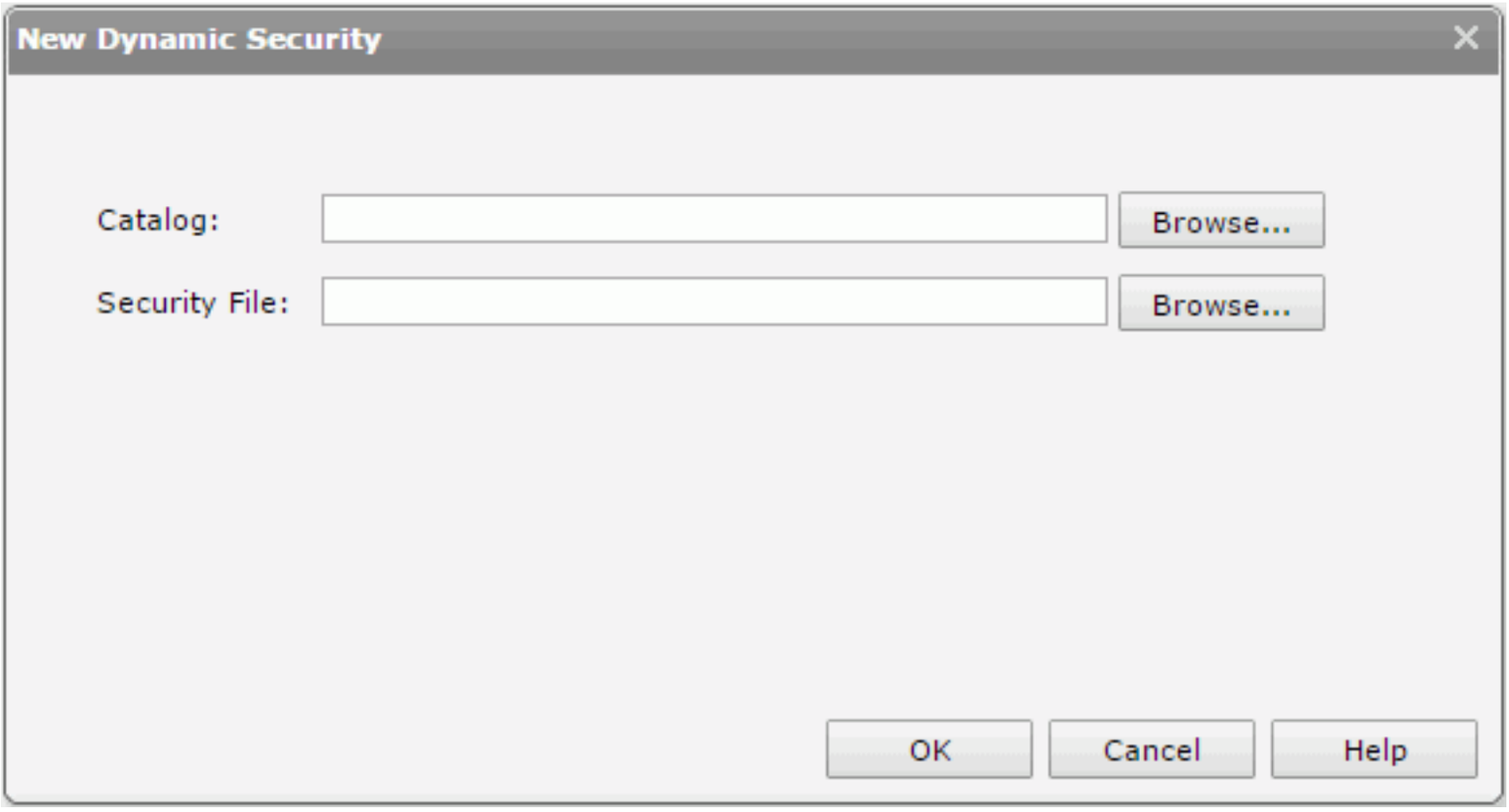
Displays the help document about this feature.

Related topics:

- Dashboard Listener API

New Dynamic Security dialog

The dialog appears when you click the New Dynamic Security link in the Dynamic Security panel. It helps you to create dynamic catalog security.



Catalog

Specifies the catalog you want to create dynamic security for.

Type the catalog with the full resource path in the text box, for example, `/SampleReports/SampleReports.cat`, or click the Browse button to select a catalog in the Select Catalog dialog.

Security File

Specifies the security file that is generated in JReport Designer and used to create dynamic security for the catalog. Click the Browse button to select a catalog security file that defines member-level security, record-level security and column-level security in the catalog.

OK

Creates the dynamic security and exits the dialog.

Cancel

Cancels the creation of the dynamic security and closes the dialog.

Help

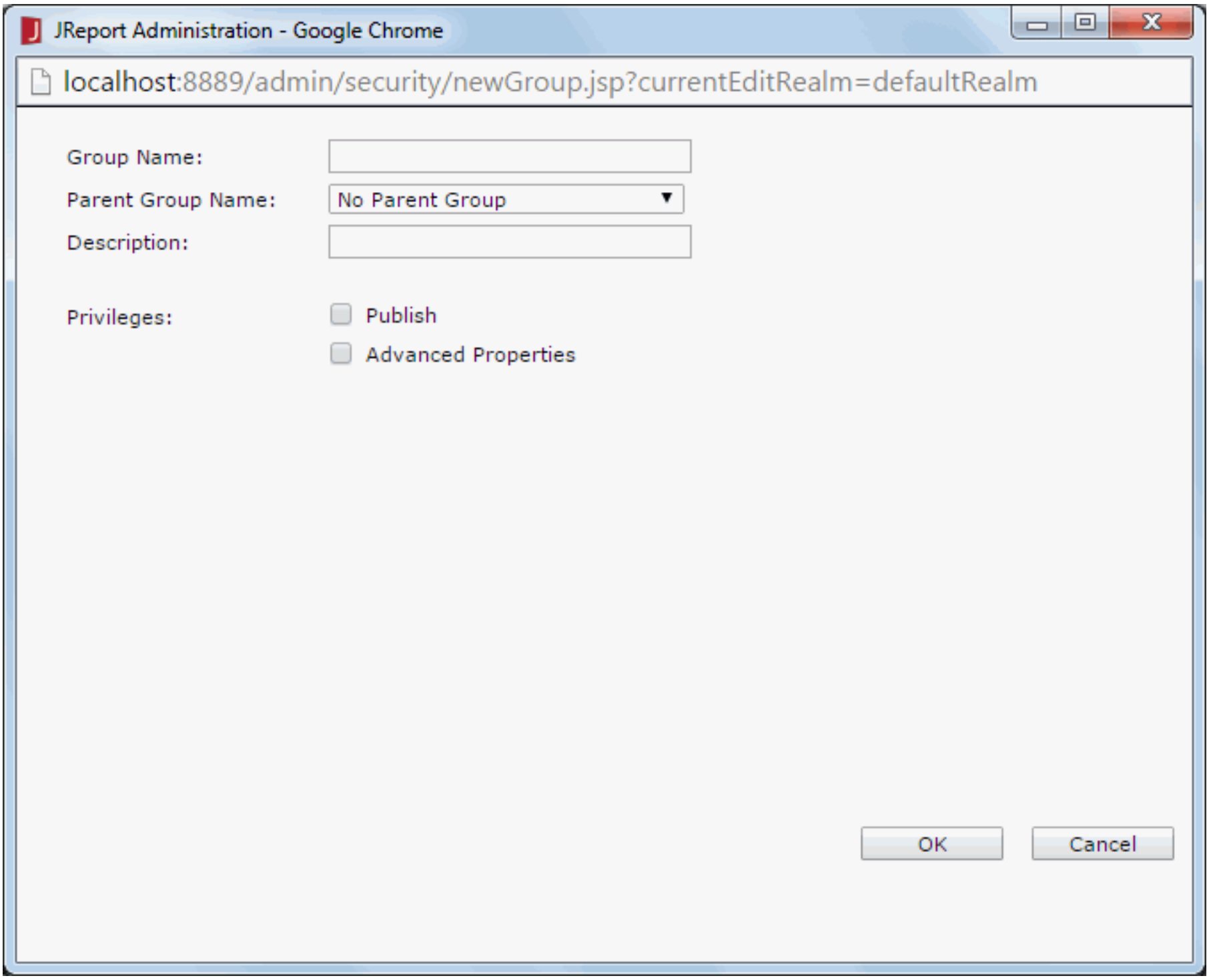
Displays the help document about this feature.

Related topics:

- Managing dynamic security

New Group dialog

This dialog appears when you click the Create a New Group link in the Security > Group panel on the JReport Administration page.



Group Name

Specifies the name for the new group.

Parent Group Name

Specifies the parent group for the new group.

Description

Specifies the description for the new group.

Privileges

Specifies whether the new group has the following privileges:

- **Publish**
  The privilege of publishing resources to JReport Server.
- **Advanced Properties**
  The privilege of viewing advanced version properties information, such as catalog connections and report related resources.

OK

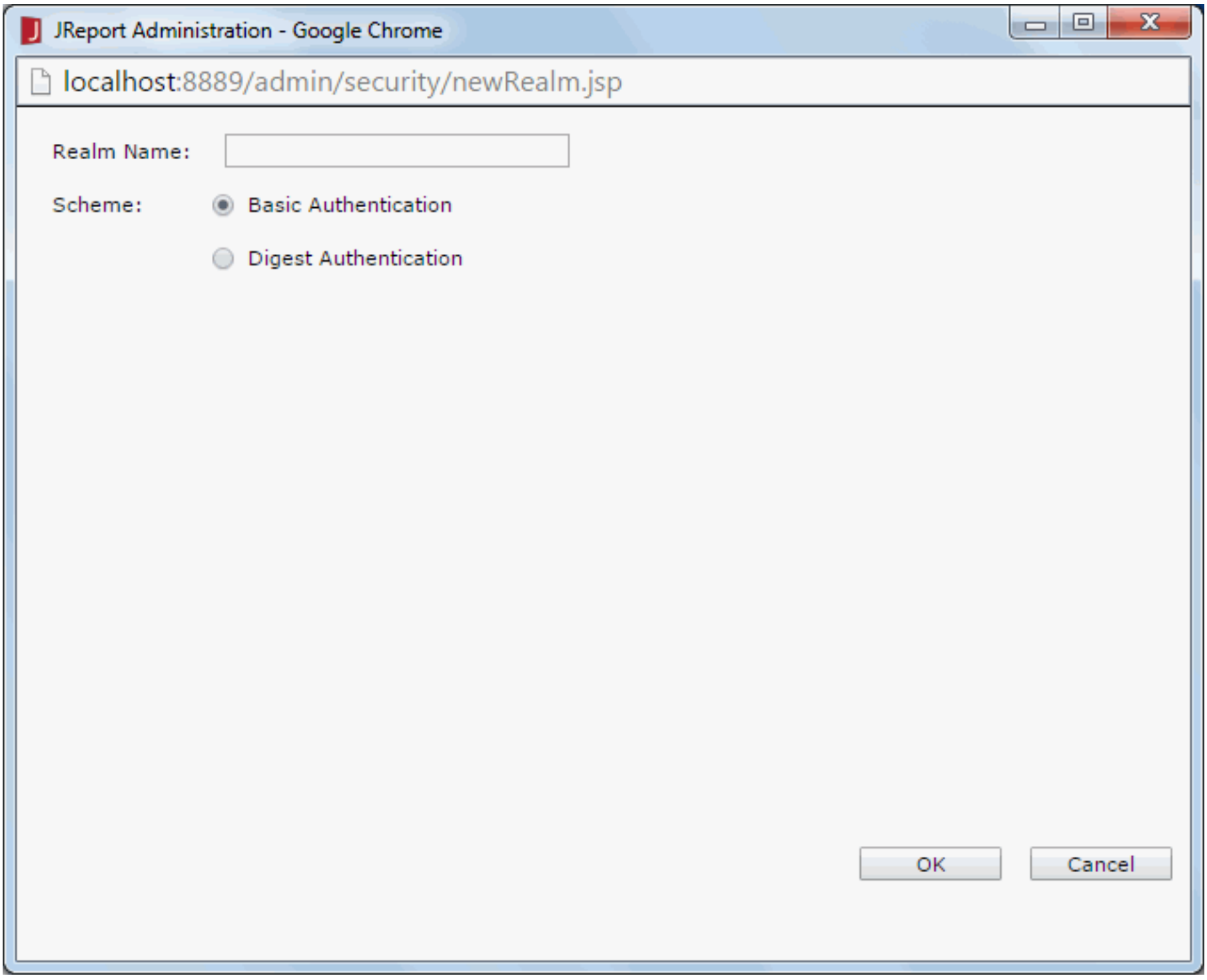
Creates a group with the information you have specified.

Cancel

Cancels operations and closes the dialog.

New Realm dialog

This dialog appears when you click the Create a New Realm link in the Security > Realm panel on the JReport Administration page.



Realm Name

Specifies the name of the new realm.

Scheme

Specifies the scheme of the new realm. It can be one of the following:

- **Basic Authentication**
  Specifies to use the Base64 method (re-conversion is easy).
- **Digest Authentication**
  Specifies to use the MD5 digest method (re-conversion is impossible).

OK

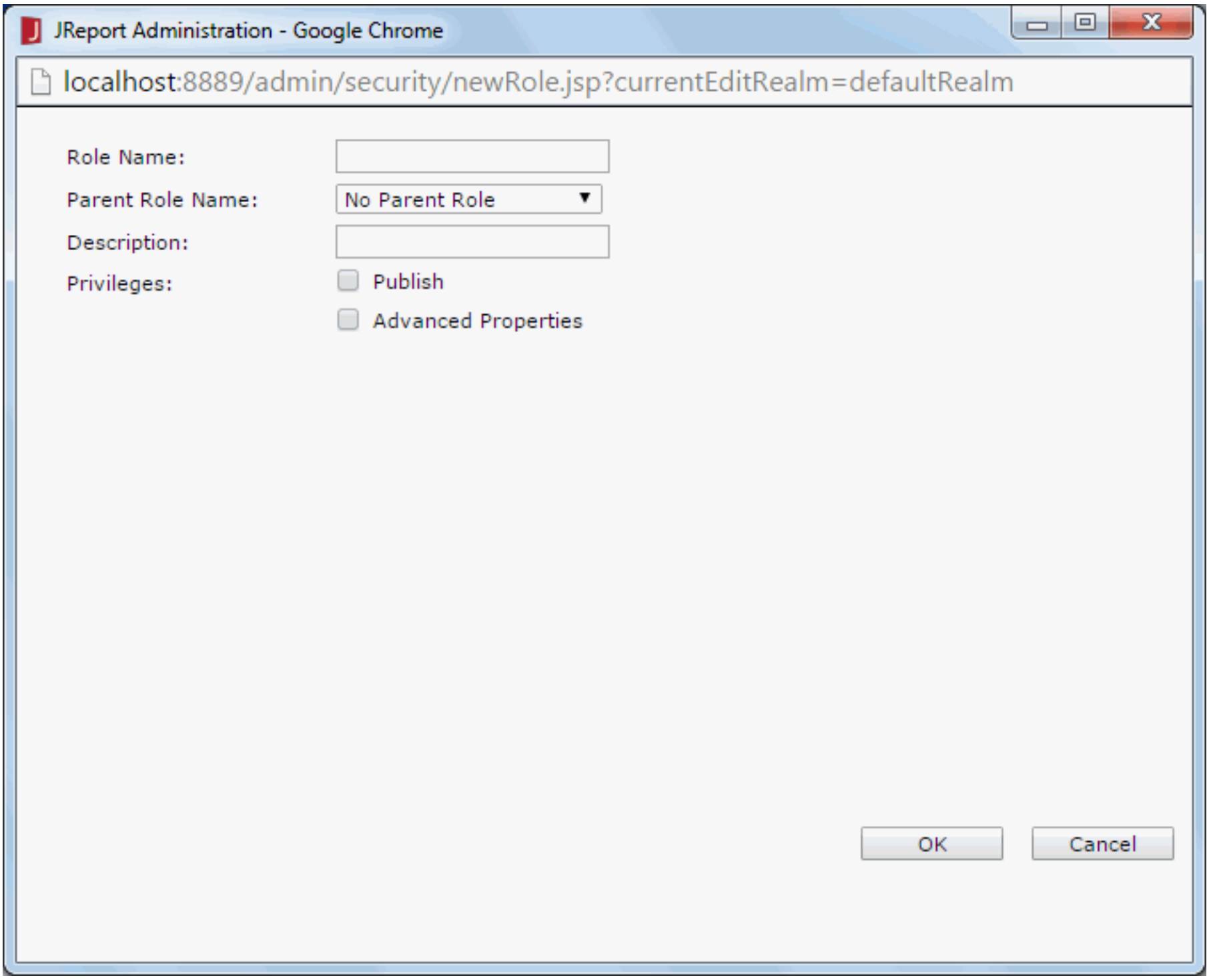
Creates a realm with the information you have specified.

Cancel

Cancels all operations and closes the dialog.

New Role dialog

This dialog appears when you click the Create a New Role link in the Security > Role panel on the JReport Administration page.



Role Name

Specifies the name for the new role.

Parent Role Name

Specifies the parent role for the new role.

Description

Specifies the description for the new role.

Privileges

Specifies whether the new role has the following privileges:

- **Publish**
  The privilege of publishing resources to JReport Server.
- **Advanced Properties**
  The privilege of viewing advanced version properties information, such as catalog connections and report related resources.

OK

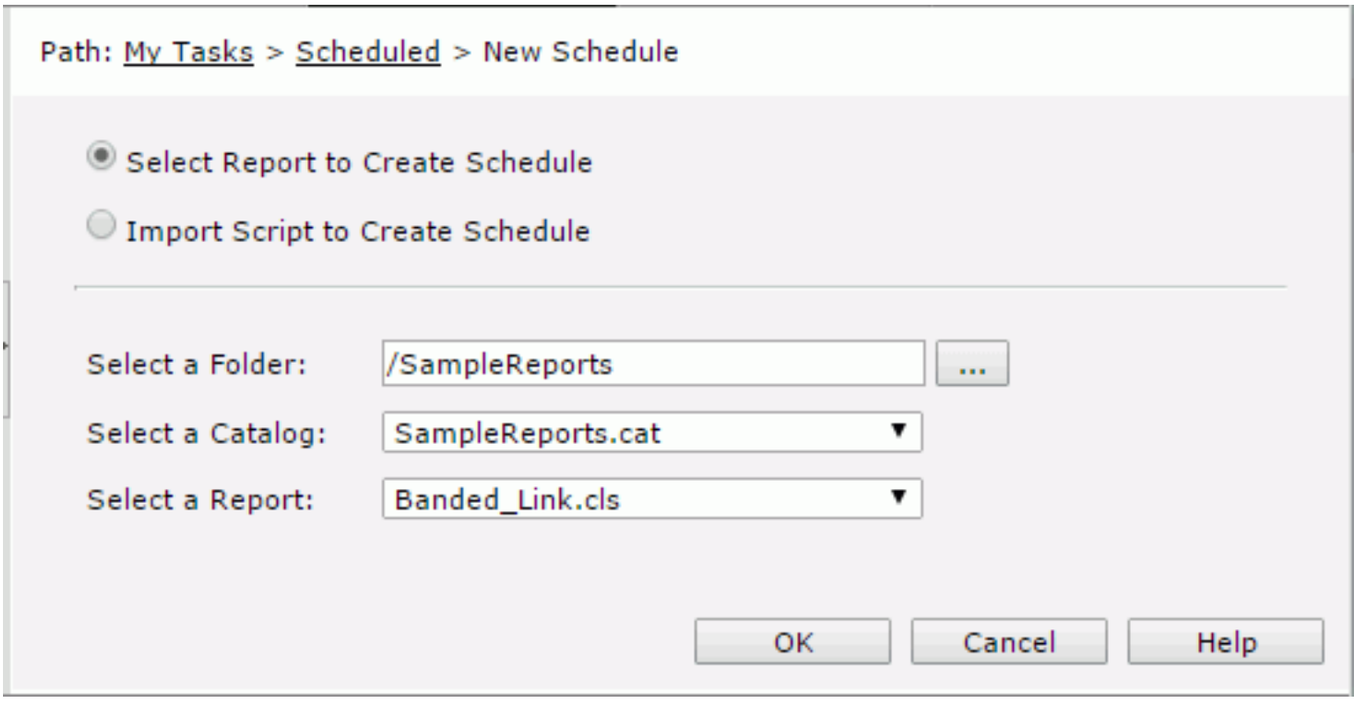
Creates a role with the information you have specified.

Cancel

Cancels operations and closes the dialog.

New Schedule dialog

This dialog appears when you click New Schedule on the task bar of the My Tasks > Scheduled page on the JReport Console page. It helps you to create a new schedule task.



Select Report to Create Schedule

Specifies to create the schedule task on an existing report.

- **Select a Folder**
  Specifies the folder in the server resource tree that contains the required report. You can click the button ... to open the Select Folder dialog for selecting the folder.
- **Select a Catalog**
  Specifies the catalog file in which the report is. Appears only after you have selected a folder first.
- **Select a Report**
  Specifies the report on which the schedule task will be created. Appears only after you have selected a folder first.

Import Script to Create Schedule

Specifies to create the schedule task on an imported script file.

- **Select Script File**
  Specifies the script file on which the schedule task will be created. Click the Browse button to select the script file.

OK

Creates the schedule task.

- If you specify to create the schedule task on an existing report, when you click OK, the Schedule dialog will be displayed. Specify the settings according to your requirement to finish the task.
- If the schedule task is created on a script file, when you click OK, the Edit Script dialog will be displayed for you to further edit the task information imported from the script file.

Cancel

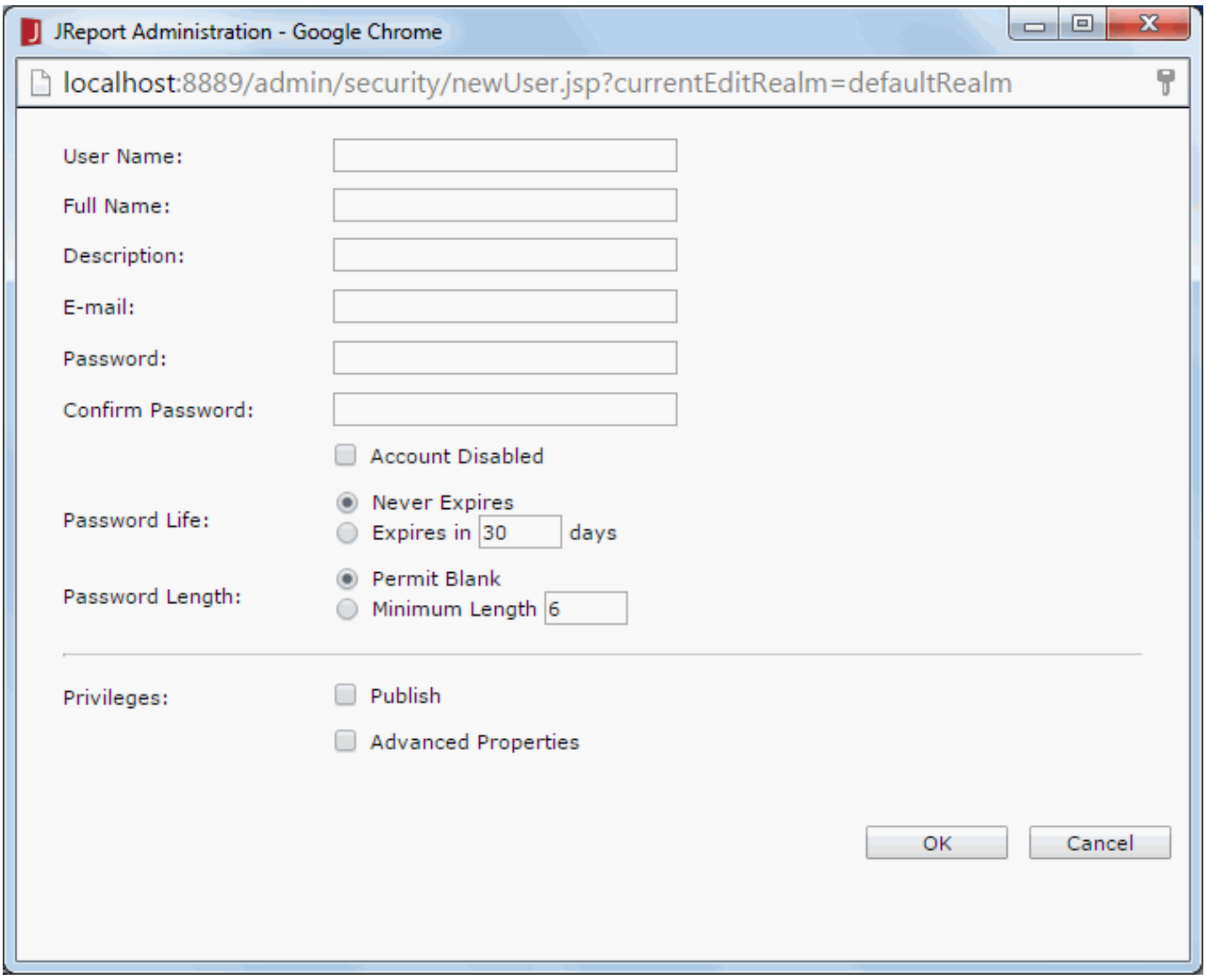
Cancels the operations.

Help

Displays the help document about this feature.

New User dialog

This dialog appears when you click the Create a New User link in the Security > User panel on the JReport Administration page.



User Name

Specifies the name for the new user.

Full Name

Specifies the full name for the new user.

Description

Specifies the description for the new user.

E-mail

Specifies the e-mail address of the new user.

Password

Specifies the password of the new user.

Confirm Password

Confirms the password by typing it in again.

Account Disabled

If this is checked, the user account will be disabled for the time being.

Password Life

Specifies the validity period of the password.

- **Never Expires**
  The password will not expire.
- **Expires in N days**
  Specifies a period of time during which the password is valid.

Password Length

Specifies the length of the password.

- **Permit Blank**
  Specifies that the password can be blank.
- **Minimum Length**
  Specifies the minimum number of characters that can be used in the password. The number you specify here should be between 0 and 20.

Privileges

Specifies whether the newly created user has the following privileges:

- **Publish**
  The privilege of publishing resources to JReport Server.
- **Advanced Properties**
  The privilege of viewing advanced version properties information, such as catalog connections and report related resources.

OK

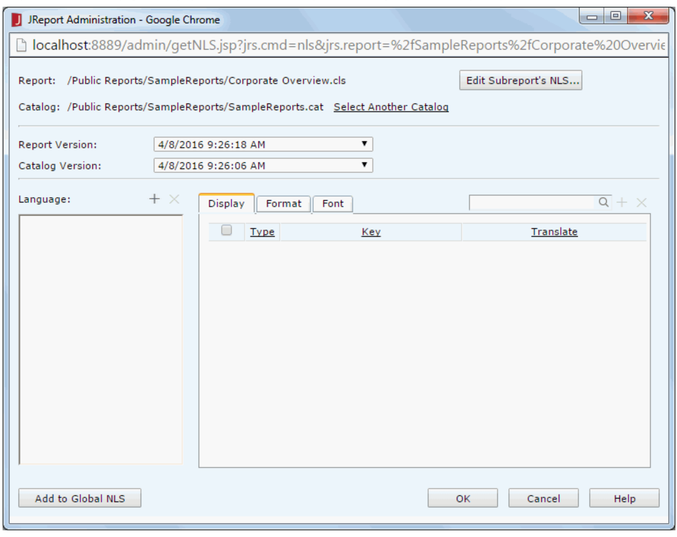
Creates the user with the information you have specified.

Cancel

Cancels operations and closes the dialog.

NLS Editor dialog

This dialog appears when you click the NLS Editor button in the Control column of a catalog, report, or library component on the JReport Administration page. It helps you to edit NLS for the specified catalog, report or library component.



Report/Component

Displays the name of the report/library component. Available only when editing NLS for a report or library component.

Edit Subreport's NLS

Specifies to edit each subreport's NLS. Available only when the specified report has subreports.

When you click this button, another dialog will appear showing all the subreports of the specified report. To edit a subreport's NLS, just click the Edit NLS link for it in the dialog. The method of editing NLS for a subreport is just the same as that for a primary report.

Catalog

Displays the name of the catalog or the catalog that the report/library component uses.

Select Another Catalog

Specifies another catalog for the report/library component in the Select Another Catalog dialog. Available only when editing NLS for a report or library component.

Report/Component Version

Specifies the report/library component version to apply the NLS settings. Available only when editing NLS for a report or library component.

Catalog Version

Specifies the catalog version.

Language

Lists the languages into which the display text in the catalog/report/library component will be translated. If you cannot find the language you need, click above the box to add it.

- When editing NLS for a report or library component, clicking the button opens the Add Language dialog to add languages.
- When editing NLS for a catalog, clicking the button opens the Select Language Source dialog to choose where to add a language.

Removes the selected language from the Language box.

Display Tab

Specifies the translation of every display text in the catalog/report/library component. You can click the underlined column header to sort the items by that column in an ascending or descending order.

- **Quick search toolbar**
  Searches for the items that contain the specified keyword. After typing the keyword, click the search button on the toolbar or press Enter on the keyboard to start searching. JReport searches in both the Key and Translate columns.
- **Type**
  Lists types of display text for different objects.
  - **Column**
    This type is only for page reports running in Page Report Studio. It is the type of display text of columns.
  - **DisplayName**
    Type of display text of object display name.
  - **Metadata**
    Type of display text of metadata. Metadata mainly refers to catalog resources, such as table/view columns, business views, formulas, summaries, parameters, and so on.
  - **Label**
    Type of display text of label, some web controls and UDOs.
  - **Prompt**
    Type of display text of parameter prompt value.
  - **Title**
    Type of display text of filter control, library component and objects in it.
  - **TOC**
    Type of display text on the TOC tree.
- **Key**
  Lists keys to indicate the objects in the original language.
- **Translate**
  Specifies the translation of the keys in the target language.
- Opens the Add Display dialog to add available display text in the objects of the catalog/report/library component for the target language.
- Removes the selected display items for the target language.

Format tab

Specifies the format for all fields in the catalog/report/library component. You can click the underlined column header to sort the items by that column in an ascending or descending order.

- **Quick search toolbar**
  Searches for the items that contain the specified keyword. After typing the keyword, click the search button on the toolbar or press Enter on the keyboard to start searching. JReport searches in both the Key and Format in <language> columns.
- **Key**
  Lists keys to indicate the formats in the original language.
- **Format**
  Specifies the formats for the keys in the target language.
- Opens the Add Format dialog to add available formats in the objects of the catalog/report/library component for the target language.
- Removes the selected format items for the target language.

Font tab

Specifies font properties for all fields in the catalog/report/library component. You can click the underlined column header to sort the items by that column in an ascending or descending order.

- **Quick search toolbar**
  Searches for the items that contain the specified keyword. After typing the keyword, click the search button on the toolbar or press Enter on the keyboard to start searching. JReport searches in both the Font Face and Font Size columns.
- **Key**
  Lists keys to indicate the fonts in the original language.
- **Font Face**
  Specifies the font faces for the keys in the target language.
- **Font Size**
  Specifies the font sizes for the keys in the target language.
- Opens the Add Font dialog to add available fonts in the objects of the catalog/report/library component for the target language.
- Removes the selected font items for the target language.

Add to Global NLS

Specifies to add the selected display/format/font items to the target language's global NLS resource library. If some of the items already exist in the global NLS resource library of the target language, the Add to Global NLS dialog appears for you to handle the duplication.

This option is not available to organization admin.

OK

Applies the settings and closes the dialog.

Cancel

Cancels the settings and exits the dialog.

Help

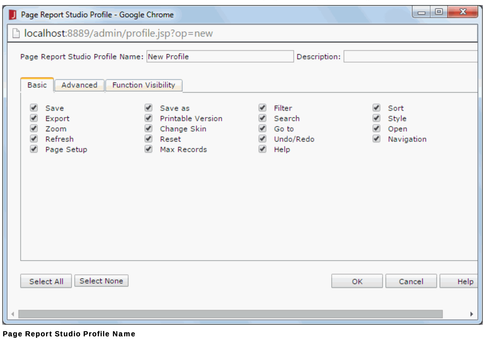
Displays the help document about this feature.

Related topics:

- Editing local NLS

Page Report Studio Profile dialog

This dialog appears when you click the New Profile link in the Profile > Customize Profile > Page Report Studio > Features tab on the iReport Administration page. It helps you to create profiles for applying in running reports in Page Report Studio.



Page Report Studio Profile Name

Specifies the name for the new profile.

Description

Specifies information about the new profile.

Basic tab

In this tab, you can turn on/off the required basic Page Report Studio features for the profile.

- **Save**
  Specifies to enable the Save feature which allows you to save the report result as a version.
- **Save as**
  Specifies to enable the Save As feature which allows you to save the current report as another file.
- **Filter**
  Specifies to enable the Filter feature, which allows you to filter report data according to your filter conditions.
- **Sort**
  Specifies to enable the Sort feature, which allows you to sort records in the data buffer by certain fields.
- **Export**
  Specifies to enable the Export feature, which allows you to export your report result for exchange purposes.
- **Printable Version**
  Specifies to enable the Print feature, which allows you to print the report data.
- **Search**
  Specifies to enable the Search feature, which allows you to search for the required data in a report.
- **Style**
  Specifies to enable the Style feature, which allows you to use styles in your report.
- **Zoom**
  Specifies to enable the Zoom feature, which allows you to zoom in on your report.
- **Change Skin**
  Specifies to enable the Change Skin feature, which allows you to change the skin of the Page Report Studio window.
- **Go to**
  Specifies to enable the Go to feature, which allows you to select to show certain groups of records in a banded object.
- **Open**
  Specifies to enable the Open Report Tab feature, which allows you to open/hide some report tabs in a report.
- **Refresh**
  Specifies to enable the Refresh feature, which allows you to reload the report data.
- **Reset**
  Specifies to enable the Reset feature, which allows you to reset your Page Report Studio window and load page defaults.
- **Undo/Redo**
  Specifies to enable the Undo/Redo feature, which allows you to undo/redo previous operations.
- **Navigation**
  Specifies to enable the Navigation feature, which allows you to navigate through the report pages.
- **Page Setup**
  Specifies to enable the Page Setup feature, which allows you to set page properties.
- **Max Records**
  Specifies to enable the Max Records feature, which allows you to set the maximum number of records retrieved by all components in the report.
- **Help**
  Specifies to enable the Help feature, which allows you to access the index page of Page Report Studio help documents with the Help button.

Advanced tab

In this tab, you can turn on/off the required advanced Page Report Studio features for the profile.

- **Query Filter**
  Specifies to enable the Query Filter feature, which allows you to apply a filter a the business view by certain component.
- **To Chart**
  Specifies to enable the To Chart feature, which allows you to turn a crosstab to a chart.
- **To Crosstab**
  Specifies to enable the To Crosstab feature, which allows you to turn a chart to a crosstab.
- **Rotate**
  Specifies to enable the Rotate feature, which allows you to rotate the selected crosstab.
- **Split**
  Specifies to enable the Split feature, which allows you to split a cell in a tabular.
- **Merge**
  Specifies to enable the Merge feature, which allows you to merge several cells in a tabular.
- **New**
  Specifies to enable the New Report feature, which allows you create a new report using a Page Report Studio window with existing queries.
- **Editing Marks**
  Specifies whether or not to enable the Editing Marks feature, which allows you to show or hide the dashed borders of some components and the report body. If the feature is disabled, the editing mark will not be shown when a report object receives focus, and report objects cannot be moved or resized.
- **Chart Type**
  Specifies to enable the Chart Type feature, which allows you to change the type of a chart.
- **Insert/Remove**
  Specifies to enable the Insert/Remove feature, which allows you to insert/remove objects in a report.
- **Grid**
  Specifies to enable the Grid feature, which helps you in laying out objects in a report.
- **Pivot**
  Specifies to enable the Pivot feature, which allows you to change the structure of a table, and crosstab, for example, you can change the order of group levels in a table/crosstab, change the order of columns in a table, and so on.
- **Resize**
  Specifies to enable the Resize feature, which allows you to resize an object.
- **Move**
  Specifies to enable the Move feature, which allows you to change the position of an object.
- **Resource View**
  Specifies to enable the Resource View feature, which allows you to insert a view element into your report.
- **Toolbox**
  Specifies to enable the Toolbox feature, which allows you to use the Toolbox to insert a component into your report.
- **Drill**
  Specifies to enable the Drill feature, which allows you to select to show certain groups of records in a crosstab.

Function Visibility tab

In this tab, you can show or hide the required Page Report Studio window elements for the profile.

- **User Information Bar**
  Specifies to enable the User Information Bar feature. With this feature enabled, you may view information about the user, catalog, and report by clicking the User Information button in Page Report Studio.
- **Page Report Bar**
  Specifies to enable the Page Report Bar feature. This feature allows you to specify whether to show the page report bar.
- **Toolbar**
  Specifies to enable the Toolbar feature.
- **Menu**
  Specifies to enable the Menu feature, which allows you to access Page Report Studio features by providing you with menu commands.
- **Table of Contents**
  Specifies to enable the Table of Contents feature, which allows you to navigate around in the report using the TOC Browser.
- **Right-click Menu**
  Specifies to enable the Right-click Menu feature. If this feature is enabled, you can further specify the commands on the right-click menu.
  - **Display Name**
    Specifies to show the object display name on the right-click menu.
  - **Filter**
    Specifies to show the Filter command on the right-click menu, which allows you to filter report data according to your filter conditions.
  - **Sort**
    Specifies to show the Sort command on the right-click menu, which allows you to sort records in the data buffer by certain fields.
  - **Go to**
    Specifies to show the Go to command on the right-click menu, which allows you to select to show certain groups of records in a banded object.
  - **Drill**
    Specifies to show the Drill command on the right-click menu, which allows you to select to show certain groups of records in a crosstab.
  - **Show/Hide**
    Specifies to show the Show/Hide command on the right-click menu, which allows you to show/hide the selected object.
  - **Rotate**
    Specifies to show the Rotate command on the right-click menu, which allows you to rotate the selected crosstab.
  - **Reset**
    Specifies to show the Reset command on the right-click menu, which allows you to reset your Page Report Studio window and load page defaults.
  - **Refresh**
    Specifies to show the Refresh command on the right-click menu, which allows you to reload the report data.
  - **Search**
    Specifies to show the Search command on the right-click menu, which allows you to search for the required information in a report.
  - **Chart Type**
    Specifies to show the Chart Type command on the right-click menu, which allows you to change the type a chart.
  - **To Chart**
    Specifies to show the To Chart command on the right-click menu, which allows you to convert a crosstab to a chart.
  - **To Crosstab**
    Specifies to show the To Crosstab command on the right-click menu, which allows you to convert a chart to a crosstab.
  - **Link Report**
    Specifies to show the Link Report command on the right-click menu, which allows you to open the linked report.
  - **Detail Report**
    Specifies to show the Detail Report command on the right-click menu, which allows you to open the detail report.
  - **Delete**
    Specifies to show the Delete command on the right-click menu, which allows you delete the selected object from a report.
  - **Query Filter**
    Specifies to show the Query Filter command on the right-click menu, which allows you to apply a filter to the business view used by the selected component.
  - **Expand Data**
    Specifies to show the Expand Data command on the right-click menu, which allows you to expand data of a crosstab or group.
  - **Properties**
    Specifies to show the Properties command on the right-click menu, which allows you to configure the properties of the selected object.
  - **Style**
    Specifies to show the Apply Style command on the right-click menu, which allows you to apply a style to the selected component.

Select All

Has all options in the tabs checked.

Select None

Has all options in the tabs unchecked.

OK

Finishes creating the profile and leaves the dialog.

Cancel

Cancels creating the profile and leaves the dialog.

Help

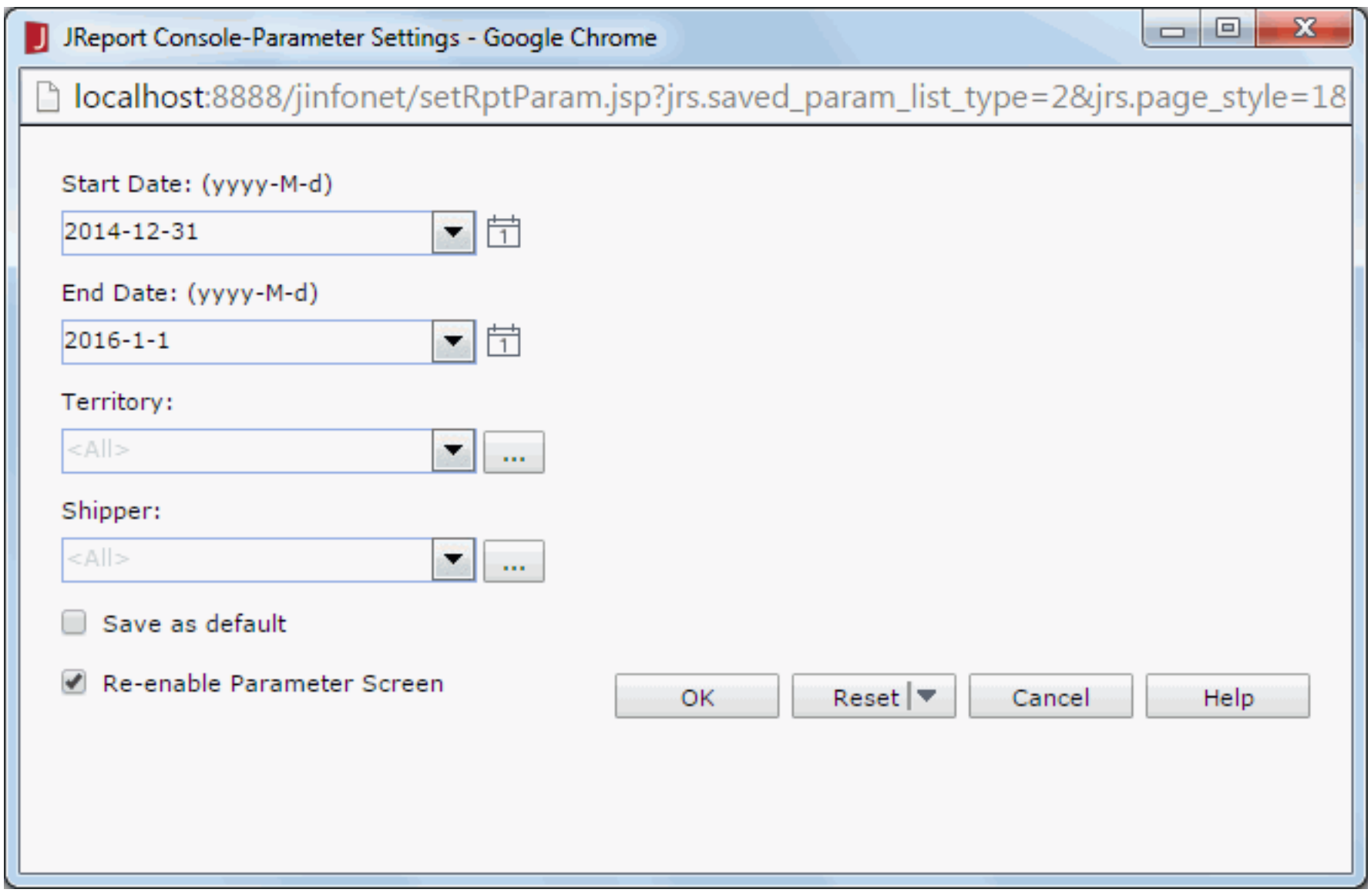
Displays the help document about this feature.

Parameter Settings dialog

The dialog allows you to customize the default parameter values for a report. It is unavailable when both Enable Setting Default Parameter Values For and Enable Hiding Initial Parameter Dialog For the corresponding resource are unchecked in the Profile > Customize Server Preferences > Advanced tab.

To access the dialog, on the JReport Console > Resources page, browse to the target report, then do one of the following:

- Select the row the report is in, then on the task bar of the Resources page, click Tools > Parameter Settings.
- Select the report row, right-click in the row and select Parameter Settings from the shortcut menu.
- Put the mouse pointer over the report row and click the Parameter Settings button on the floating toolbar.



Parameters

Displays parameters of the report with your last-time saved default values, which could be the values saved in this dialog last time, or in the Enter Parameter Values dialog, or when advanced running or scheduling the report, and for a web report, could also be the values saved in the Parameters panel. Edit the values according to your requirement. You may specify parameter values in these ways.

If you have not yet set the default values on the server, or if you did but your saved default values cannot fully match the current parameters, all the parameters will use their default values specified in the parameters' definition as the initial values.

If no parameter is used in the report, "No Parameter Needed" will be displayed here.

Save as default

Saves current parameter values as the default values for the report. It is a user-report level setting.

This option is an action and it takes effect after you click OK in the dialog. Its initial status is always unselected.

For page reports, this option is enabled when Enable Setting Default Parameter Values For Page Report is checked in the Profile > Customize Server Preferences > Advanced tab.

For web reports, this option is enabled when Enable Setting Default Parameter Values For Web Report is checked in the Profile > Customize Server Preferences > Advanced tab.

Re-enable Parameter Screen

When this option is unselected, the next time the report runs via operation on the console, it will use the last-time saved default parameter values directly without popping up the Enter Parameter Values dialog. However, if the last-time saved default values cannot completely match the report parameters, the dialog will still be displayed. It is a user-report level setting.

You can check it to show the Enter Parameter Values dialog again after it has been hidden.

For page reports, this option is enabled when Enable Hiding Initial Parameter Dialog For Page Report is checked in the Profile > Customize Server Preferences > Advanced tab.

For web reports, this option is enabled when Enable Hiding Initial Parameter Dialog For Web Report is checked in the Profile > Customize Server Preferences > Advanced tab.

OK

Applies the settings and exits the dialog.

Reset

When Save as default is enabled in the dialog, the Reset button contains a text part and a triangle icon. You can choose to reset the values to either of the following by clicking the triangle. If you click the text part of the button directly, the values will be reset to the original default values.

- **Original Default Values**
  The default values defined in the parameters' definition.
- **User Defined Default Values**
  Your last-time saved default values.

When Save as default is disabled in the dialog, the Reset button contains only the text part. Clicking it will reset the values to the original default values.

Cancel

Closes the dialog and discards any changes.

Help

Displays the help document about this feature.

Note: If you just edit the parameter values in the dialog and click OK, the new values will be meaningless. Only when you check Save as default and click OK, can the values be saved as default values to the report.

Profile dialog

The Profile dialog is available on both the JReport Administration and Console pages. The difference lies in that any settings you specify in the Profile dialog of the JReport Console page will only be applied to the current user, while the settings specified on the JReport Administration page will be applied to all users. However, the profile settings made on the Administration page will not be able to apply to the Console page whose profile settings have already been saved by clicking the OK button in the Profile dialog. In this case, if the admin would like the Console page to adopt the same change as the Administration page, the admin should ask the corresponding end user to make changes on the Console page directly.

To access this dialog, click Profile on the system toolbar. The dialog contains three subjects:

- Change Password
- Customize Server Preferences
- Customize Profile

Reset

Resets the settings without refreshing the web pages.

Restore Defaults

Uses the default settings. Available only when the Profile dialog is accessed on the JReport Console page.

OK

Submits any changes to JReport Server to allow the customizing of your web pages.

Cancel

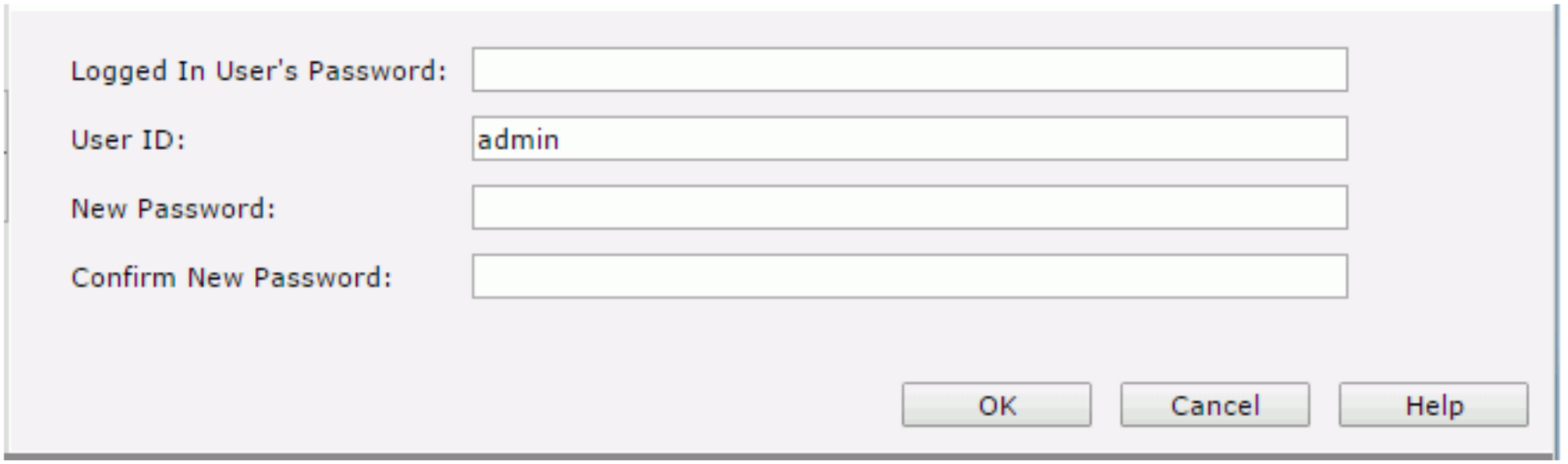
Cancels the settings.

Help

Displays the help document about this feature.

Change Password

This panel allows you to change the password used for logging onto JReport Server. It is available only when the Profile dialog is accessed on the JReport Console page.



Logged In User's Password

Specifies the password of the currently logged in user.

User ID

Displays the ID of the current user.

New Password

Specifies the new password.

Confirm New Password

Confirms the new password by typing in it again.

Customize Server Preferences

This panel allows you to customize your JReport Server web pages. It contains the following tabs:

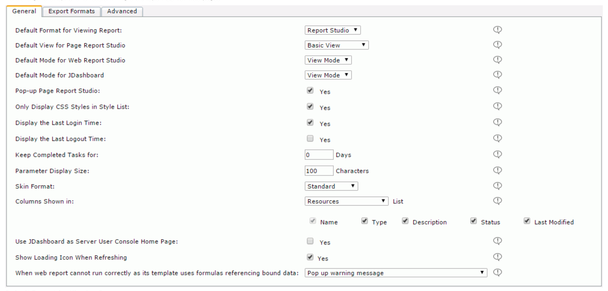
- General
- Export Formats
- Advanced

Allow users to modify preferences from user console

Specifies whether to allow end users to modify the server preferences from the JReport Console page. Available to administrators only.

General

Lists the general options for customizing the JReport Server web pages.



Default Format for Viewing Report

Specifies the default format for viewing reports. If Report Studio is selected, page reports will be opened in Page Report Studio, and web reports in Web Report Studio.

If the property Default Format for Viewing Report of a page report tab or web report has been set to a specific format when the report is designed in JReport Designer, the value in Designer has higher priority than the one here.

Default View for Page Report Studio

Specifies the default mode for viewing page reports in Page Report Studio. You can choose between Basic View and Interactive View.

Default Mode for Web Report Studio

Specifies the default mode for viewing web reports in Web Report Studio. You can choose between View Mode and Edit Mode.

Default Mode for JDashboard

Specifies the default mode for viewing dashboards in JDashboard. You can choose between View Mode and Edit Mode.

Pop-up Page Report Studio

Specifies whether to show a Page Report Studio window when you view a report in Page Report format.

Only Display CSS Styles in Style List

Specifies whether to display only the CSS styles in the style list.

Display the Last Login Time

Specifies whether to display the last login time of a user on the top banner of JReport Server. If checked, the login time will be recorded in the login.properties file in `<install_root>\bin` after JReport Server shuts down.

Display the Last Logout Time

Specifies whether to display the last logout time of a user on the top banner of JReport Server. If checked, the logout time will be recorded in the logout.properties file in `<install_root>\bin` after JReport Server shuts down.

Keep Completed Tasks for

Specifies for how many days JReport Server will keep the completed tasks in the Completed list. If the value is set to 0, the server will keep the completed tasks until you delete them from the Completed page.

Parameter Display Size

Specifies the display length in characters of the parameters that are to be applied to both the completed and active schedules.

Folder Selector Type

Specifies the type of the tool used for selecting folder paths from the file system. This option is available on the Internet Explorer browser and applied to local publishment only.

There are three types of folder selectors:

- **JavaScript format**
  Folder selector is a dialog generated by JavaScript.
- **VBScript format**
  Folder selector is a dialog generated by VBScript.
- **JSP format**
  Folder selector is an HTML page generated by JSP. This format is available to administrators only.

Skin Format

Specifies the skin format applied to JReport Server and Page Report Studio UI. You can choose from Standard, Classical and Windows XP.

Columns Shown in _ List

Specifies which columns will be displayed as default in the following list when you log onto JReport Server: Resources, Scheduled, Running, Completed, Background Tasks, Report Result Versions, Report Versions, Catalog Versions, and Result Versions.

Use JDashboard as Server User Console Home Page

Specifies whether to enable dashboard users to set JDashboard status as the server home page, this provides much convenience when dashboard users need to visit some dashboards a lot. See Setting JDashboard as the server home page for details.

Show Loading Icon When Refreshing

Specifies whether to show the loading icon when refreshing reports or dashboards.

Enable IE Navigate2

Specifies to load resources with IE Navigate2 method and its flags setting when viewing reports or loading URL links in Internet Explorer.

When web report cannot run correctly as its template uses formulas referencing bound data

Specifies what to do when a web report cannot be created with the selected web report template, which uses a formula that references bound data to control the company logo. You can choose to stop creating the report, to remove error information and continue creating the report, or to pop up warning message. When you select to remove error information and continue creating the report, all the data context like the bound data and formula will be removed from the template.

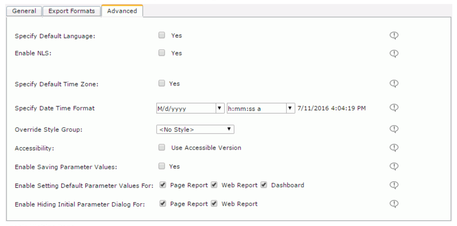
Export Formats

Customizes the export formats.

- **JReport Result**
  Exports a report to a result file. Once this option is checked, the following option is available:
  - **Zip Result**
    Specifies whether to compress the result and make its size smaller.
- **Page Report Result**
  - **Resolution**
    Specifies the resolution of the result to zoom in/out, in DPI. The default value is obtained from the operation system, which is the resolution of your monitor, for example, 72 DPI on Unix or 96 on Windows. You can set higher/lower value to zoom in/out.
- HTML
- PDF
  - **Clipping Area**
    Specifies whether to keep the clipping area of the report in the exported PDF file.
- Excel
- Text
- RTF
- XML
- PostScript
- Applet

Advanced

Lists the advance options for customizing the JReport Server web pages.



Enable Waiting Page

Specifies whether to show the waiting pages. Click the Customize button to customize waiting pages in the Customize Waiting Page dialog. This option is available only when the Profile dialog is accessed on the JReport Administration page.

Specify Default Language

Specifies an environment language for JReport Server. All UI text and messages will be displayed in the specified language. The available languages come from the qualified language packages defined in `<server_install_root>\resources\server\languages`. By default there are two built-in English language packages: en and en-us. You can add more language packages following the section NLS at application level.

- **Reset All Users' Preference**
  Checking this option will reset all users' default language to the one specified here.

Enable NLS

Specifies to enable the NLS feature when directly run reports and dashboards. This option is available only when the Profile dialog is accessed on the JReport Console page.

- **Default Language**
  Specifies the default language in which the report and dashboard results are displayed.
- **Default Encoding**
  Specifies the default encoding for the reports and dashboards.

Specify Default Time Zone

Specifies the default time zone for JReport Server. All date on the server UI will then be displayed according to the specified time zone.

Specify Date Time Format

Specifies the default date and time formats in which such values will be displayed on the JReport Server web pages, for example, the last modified time of a report, the time when a schedule take is finished.

Override Style Group

Specifies the default style group that will be applied when running reports on JReport Server, including direct running, advanced running and scheduled running. Reports running in Page Report Studio will also be exported with this default style group if no other style group has been specified.

Identify Server Preference

Specifies whether a task can be performed on a manually specified clustered server when scheduling a report. This option is available only in cluster environment.

Use Accessible Version

Sets the accessible version of JReport Server as the default portal instead of JReport Console page. In the accessible version, reports are displayed in the HTML format with accessibility attributes, and table/crosstab components are output as HTML data table. With the help of reader agent end users can navigate through the server resource to view the target report. See Accessibility for more information.

Enable Saving Parameter Values

Enables saving parameter values for reuse next time when specifying parameter values in order to run reports.

For a report that has multiple parameters, the values of all the parameters specified at a time will be saved together as a group for the report. Here such group is also called a list. Selecting such a group as the value will apply the values in the group to all the corresponding parameters in the report.

When the Yes checkbox is selected, you need to choose a way of saving the parameter values: Manually or Automatically. When switching from one to the other and OK is clicked, all saved parameter values for the current user will be cleared.

- **Manually**
  If checked, it is up to users to manually save the parameter values they are going to reuse frequently and manage the saved values. Then in some places where users are asked to specify parameter values, such as advanced running and scheduled running, the options for using saved parameter values, saving the specified values, and deleting the saved values will be provided.
- **Automatically**
  If checked, JReport saves user submitted parameter values on a user-report or user-dashboard basis.

  The automatically saved parameter values groups will be available on the value drop-down list of each of those parameters next time when the same user is asked to specify parameter values for the same report. Selecting a group from any of the parameters' value lists will apply the saved group of values to all the parameters. A group on the value lists is a one-row values' set: it lists the values of the parameters in one row with blank to separate between parameters and the values are in an order same as the display order of the parameters.
- **Maximum Number of Auto Complete Parameters List**
  For both manually and automatically saved parameter values groups or lists for a report, there is a number restriction. By default it is 3, which means the manually saved lists for a report or the auto saved lists for a report can be no more than three. When the saved lists exceed the maximum number or if you change the current number to a smaller one, the oldest lists will be removed.

Enable Setting Default Parameter Values For

Specifies whether to display the Save as default option on the parameter page which allows server users to customize the default parameter values for reports or dashboards on a user-report or user-dashboard basis. For details, see Customizing default parameter values.

- **Page Report**
  Enables users to set default parameter values for page reports.
- **Web Report**
  Enables users to set default parameter values for web reports.
- **Dashboard**
  Enables users to set default parameter values for dashboards in JDashboard.

Enable Hiding Initial Parameter Dialog For

Specifies whether to display the Do not show the screen again option in the Enter Parameter Values dialog or the Re-enable Parameter Screen option in the Parameter Settings dialog or Parameter panel in Web Report Studio, which allows server users to hide the initial parameter dialog brought by actions on the console to run a report on a user-report basis.

- **Page Report**
  Enables users to hide the initial parameter dialog for running page reports.
- **Web Report**
  Enables users to hide the initial parameter dialog for running web reports.

Customize Profile

This panel allows you to customize Page Report Studio, Web Reports and Dashboard features according to your requirement. It contains three tabs:

- Location
- Page Report Studio
- Dashboard

Location

This panel allows you to define some settings that will be applied to Page Report Studio, Web Report Studio and Dashboard.

Specify Date Time Format

Specify the date and time format (or format) to which such values will be displayed in the Page Report Studio, Web Report Studio and Dashboard with pages. In addition, the last modified time on print link of a report, the date values displayed in the query filter area etc.

- **Date Format**
  Specifies the default format for the date values.
- **Time Format**
  Specifies the default format for the time values.
- **Timeline Format**
  Specifies the default format for the timeline values.

Sort on Column Headers

Specifies to allow the end-user to sort the data values by clicking the sort button on table headers in Web Report Studio and Dashboard.

Filter on Column Headers

Specifies to allow the end-user to filter the data values by clicking the filter button on table headers in Web Report Studio and Dashboard.

Show Sort/Filter Status on Column Headers

If checked, the sort/filter icon will always be displayed on the table column headers with the current sort/filter status. If the checkbox is unchecked, and users hover over the table column headers, the sort/filter icon will appear. The default checkbox state is unchecked.

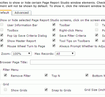
Page Report Studio

This tab allows you to customize Page Report Studio features according to your requirement. It contains two sub tabs:

- Palette
- Properties

Palette

Administrators can define icons their own Page Report Studio Palette profile in the *Report Administration* page. Here and users can select one of the profiles to apply for running reports in Page Report Studio.



Default Profile

Specifies the profile which will be used as the default profile to run reports in Page Report Studio. End users can only select a profile and view its settings.

Save Profile

Opens the **Save Palette** dialog box allowing users to create profiles in applying running reports in Page Report Studio. Available to administrators only.

Delete Profile

Deletes the selected profile. Available to administrators only.

Profile list

The list of Page Report Studio profiles that have been created:

- **Name**
  Displays the names of the profiles.
- **Description**
  Displays the descriptions of the profiles.
- **Last Modified**
  Displays the time when the profiles were last modified.

Properties

This tab lists the properties for configuring the Page Report Studio settings. The properties are classified in two categories: default properties and advanced properties. Some properties are not available when the Profile dialog is accessed on the *Report Creation Console* page. If you want to customize them, log onto the Report Administration page.

Default Properties

Specifies the profile which will be used as the default profile to run reports in Page Report Studio. End users can only select a profile and view its settings.

Save Profile

Opens the **Save Palette** dialog box allowing users to create profiles in applying running reports in Page Report Studio. Available to administrators only.

Delete Profile

Deletes the selected profile. Available to administrators only.

Profile list

The list of Page Report Studio profiles that have been created:

- **Name**
  Displays the names of the profiles.
- **Description**
  Displays the descriptions of the profiles.
- **Last Modified**
  Displays the time when the profiles were last modified.

Properties

This tab lists the properties for configuring the Page Report Studio settings. The properties are classified in two categories: default properties and advanced properties. Some properties are not available when the Profile dialog is accessed on the *Report Creation Console* page. If you want to customize them, log onto the Report Administration page.

Default Properties

Specifies the profile which will be used as the default profile to run reports in Page Report Studio. End users can only select a profile and view its settings.

Save Profile

Opens the **Save Palette** dialog box allowing users to create profiles in applying running reports in Page Report Studio. Available to administrators only.

Delete Profile

Deletes the selected profile. Available to administrators only.

Profile list

The list of Page Report Studio profiles that have been created:

- **Name**
  Displays the names of the profiles.
- **Description**
  Displays the descriptions of the profiles.
- **Last Modified**
  Displays the time when the profiles were last modified.

This table lists the properties for configuring the Page Report Studio dashboard elements. The properties are classified by two working modes of Dashboard: Edit Mode and Show Mode.

Identical to Dashboard Edit Mode, Show Mode provides a dashboard for view function, whether the corresponding function is enabled or the current profile and whether the option is available from a list for the properties tab. Only when both states are enabled can the option be available for use. If a feature is disabled in Edit Mode, the corresponding function in Show Mode will be disabled, regardless of the corresponding function is enabled or not in Show Mode.

Edit Mode

Specifies to show or hide some Dashboard Edit Mode elements in the edit mode. Select/unselect the element names to show/hide them. If unselected, the corresponding elements will be hidden in Edit Mode.

Show Toolbar

Toolbar Icons specifies which commands the menu options for working with dashboards.

- **Toolbar Icons** specifies which icons on the toolbar. The following options are enabled when Show Toolbar is selected:
  - **Refresh**
    Refreshes the current dashboard
  - **Enter Parameter Values**
    Lists all the parameters used in the dashboard for specifying their values.
  - **Export**
    Exports the current dashboard.
  - **Print**
    Prints the components in the current dashboard.
  - **Clear Filters**
    Removes all the filters from the current dashboard including those generated via sliders, filter controls, messages, drilling and going actions, and those assigned using web browsers such as Page Report Studio and Web Report Studio, except query filters and others designed and saving effect in Report Studio.
  - **Settings**
    Displays the Dashboard Settings dialog box.
- **Show Dashboard Header**
  Displays the current dashboard header. Force being shown is hidden or force being hidden is shown.
- **Show Dashboard Footer**
  Displays the current dashboard footer. Force being shown is hidden or force being hidden is shown.
- **Help**
  Displays the Dashboard Help documents.

Options Menu

Specifies the options available on the menu after clicking ⋮ on the toolbar of Dashboard. The following options are enabled when Show Toolbar is unselected and Show Toolbar Icons section is selected:

- **New**
  Creates a new dashboard
- **Open**
  Opens a dashboard to open
- **Save**
  Saves the changes made to the current dashboard
- **Save As**
  Saves the current dashboard with a different name or to a new location.
- **Export**
  Exports the components in the current dashboard
- **Print**
  Prints the components in the current dashboard
- **Clear Filters**
  Removes all the filters from the current dashboard including those generated via sliders, filter controls, messages, drilling and going actions, and those assigned using web browsers such as Page Report Studio and Web Report Studio, except query filters and others designed and saving effect in Report Studio.
- **Arrange**
  Arrangements arranges the library components in the current dashboard to be tidy once a key is chosen. It is an instant action.
- **Edit Active Header**
  Opens the header section of the active page in the Options Menu section.
- **Edit Active Footer**
  Opens the footer section of the active page in the Options Menu section.
- **Change Parameter Values**
  Opens the Change Parameter Values dialog box.
- **Change Page Layout**
  Arrangements arranges the library components in the current dashboard to be tidy once a key is chosen. It is an instant action.
- **Change Page Size**
  Changes the page size for displaying the items selected in the Options Menu section below.
- **Change Page Orientation**
  Changes the page orientation for displaying the items selected in the Options Menu section below.
- **Hide Dashboard Header**
  Hides the current dashboard header. Force being shown is hidden or force being hidden is shown.
- **Hide Dashboard Footer**
  Hides the current dashboard footer. Force being shown is hidden or force being hidden is shown.
- **Help**
  Displays the Dashboard Help documents.
- **Edit**
  Opens the Dashboard Edit Mode.
- **Theme**
  Specifies the theme applied to dashboards. This option is enabled when Theme in the Options Menu section is selected.
- **Component Options Menu**
  Specifies the options available on the menu after clicking ⋮ on the title bar of a library component. The following options are enabled when Show Toolbar is selected:
  - **Edit Setting**
    Opens the setting panel
  - **Export**
    Exports the library component.
  - **About**
    Displays the about information such as ID, the author and the time in mail address, and the description about the component. By default, this is disabled.

Show Toolbar

Specifies to show or hide action (Dashboard window elements in the view mode. Select/unselect the element names to show/hide them.

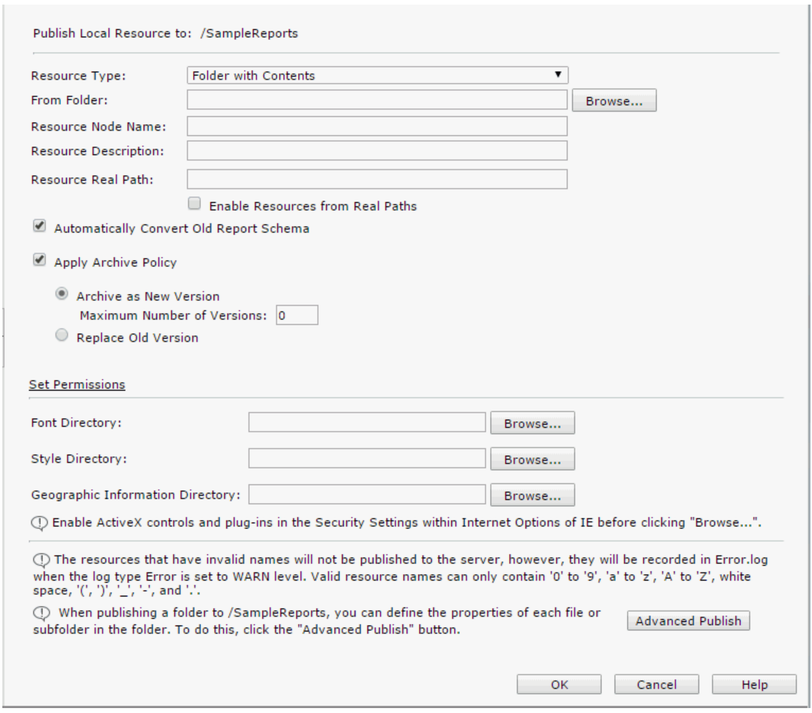
Show Toolbar

Toolbar Icons specifies which commands the menu options for working with dashboards.

- **Toolbar Icons** specifies which icons on the toolbar. The following options are enabled when Show Toolbar is selected:
  - **Refresh**
    Refreshes the current dashboard
  - **Enter Parameter Values**
    Lists all the parameters used in the dashboard for specifying their values.
  - **Export**
    Exports the current dashboard.
  - **Print**
    Prints the components in the current dashboard.
  - **Clear Filters**
    Removes all the filters from the current dashboard including those generated via sliders, filter controls, messages, drilling and going actions, and those assigned using web browsers such as Page Report Studio and Web Report Studio, except query filters and others designed and saving effect in Report Studio.
  - **Settings**
    Displays the Dashboard Settings dialog box.
- **Hide Dashboard Header**
  Displays the current dashboard header. Force being shown is hidden or force being hidden is shown.
- **Help**
  Displays the Dashboard Help documents.
- **Edit**
  Opens the Dashboard Edit Mode.
- **Theme**
  Specifies the theme applied to dashboards. This option is enabled when Theme in the Options Menu section is selected.
- **Component Options Menu**
  Specifies the options available on the menu after clicking ⋮ on the title bar of a library component. The following options are enabled when Show Toolbar is selected:
  - **Edit Setting**
    Opens the setting panel
  - **Export**
    Exports the library component.
  - **About**
    Displays the about information such as ID, the author and the time in mail address, and the description about the component. By default, this is disabled.

Publish to Local Server dialog

The Publish to Local Server dialog guides you through the process of publishing resources to JReport Server locally.



Publish Local Resource to

Shows where the resource will be published to.

Resource Type

Specifies the type of the resource.

From Folder/From File

Specifies where to get the resource. Click the Browse button to specify the location.

Resource Node Name

Specifies the name of the resource. This name is required and is used as the display name of the resource in the server resource tree.

Resource Description

Specifies the description of the resource.

Status

Specifies the status of the report. If not specified, the status will be Active by default. This option is available for report type only.

- **Active**
  The report can be run, advanced run and scheduled on JReport Server.
- **Inactive**
  The report cannot be run, advanced run or scheduled on JReport Server.
- **Incomplete**
  The report is not completely designed and cannot be run, advanced run or scheduled JReport Server.

Resource Real Path

Specifies the real path of the folder. This option is available for folder type only.

Enable Resources from Real Paths

Specifies whether or not to enable getting resources from the folder's real path. This option is available for folder type only. Once enabled, the real path resources will be mapped into the resource node of the folder in the server resource tree and the server will always be able to get the resources and updates from the real path. For more information, refer to Getting and using resources from a real path.

Note: This option can only be checked when the Enable Resources from Real Paths option in the Configuration > Advanced panel on the JReport Administration page has also been checked.

[Custom Field Name]

Specifies value of the custom field for the resource. A custom field can be regarded as a resource property and is available when it is enabled.

Automatically Convert Old Report Schema

Specifies whether or not JReport Server converts JReport reports of earlier versions into current version JReport reports when publishing the earlier version reports.

Apply Archive Policy

Applies an archive policy to the resource versions.

- **Archive as New Version**
  Specifies whether to use multiple versions for the resource.
  - **Maximum Number of Versions**
    Specifies the maximum number of versions that will be listed in the version table of the resource. The default value is 0, which means that the version number is unlimited.
- **Replace Old Version**
  Specifies to replace the old version when the new version is generated.

Set Permissions

Available only when the resource is to be published to the Public Reports or Public Components folder. Click the link to set user permissions to the resource in the Set Permissions dialog.

Font Directory

Specifies the font directory of the resource. Click the Browse button to specify the directory. Note that only the font used by the specified resource can be published.

Style Directory

Specifies the style directory of the resource. Click the Browse button to specify the directory.

Geographic Information Directory

Specifies the geographic information of the resource with reports or library components contained that have geographic information. Click the Browse button to specify the directory.

Advanced Publish

Specifies to publish resource in an advanced way.

- **Checkbox**
  Specifies the resources you want to publish.
- **File Name**
  Displays the file name of the resource.
- **Resource Name**
  Specifies the name of the resource. This name is required and is used as the display name of the resource node in the server resource tree.
- **Description**
  Specifies the description of the resource.
- **[Custom Field Name]**
  Specifies value of the custom field for the resource.
- **Advanced**
  Specifies the advanced properties for the resource, which include the archive policy of the resource, and the user permissions on the resource if the resource is to be published to the Public Reports folder.

OK

Retains the settings and submits the task to server.

Cancel

Cancels the operations.

Help

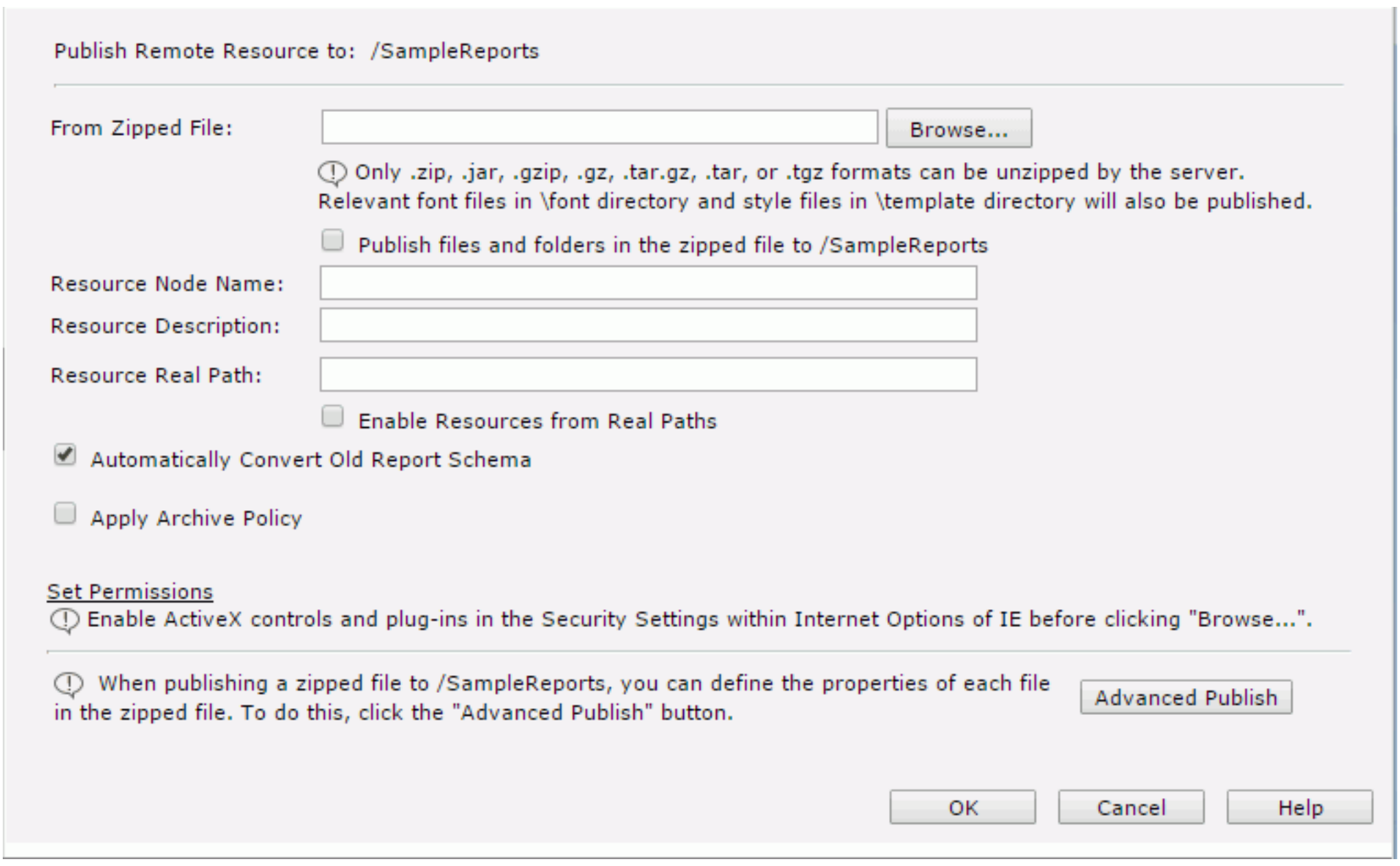
Displays the help document about this feature.

Related topics:

- Publishing resources

Publish to Remote Server dialog

The Publish to Remote Server dialog guides you through publishing resources to JReport Server remotely.



Publish Remote Resource to

Shows where the resources will be published.

From Zipped File

Specifies where to get the resources. Click the Browse button to specify the location.

Publish files and folders in the zipped file to /XXX

Specifies whether to publish files and folders in the zipped file to /XXX directly.

When this option is not checked, the following options are available, which specify properties of the new folder you want to create in /XXX to locate the resources in the zip file.

- **Resource Node Name**
  Specifies the node name of the folder. This name is required and is used as the display name of the folder in the server resource tree.
- **Resource Description**
  Specifies the description of the folder.
- **Resource Real Path**
  Specifies the real path of the folder.
- **Enable Resources from Real Paths**
  Specifies whether or not to enable getting resources from the folder's real path. Once enabled, the real path resources will be mapped into the resource node of the folder in the server resource tree and the server will always be able to get the resources and updates from the real path. For more information, refer to Getting and using resources from a real path.

  Note: This option can only be checked when the Enable Resources from Real Paths option in the Configuration > Advanced panel on the JReport Administration page has also been checked.
- **[Custom Field Name]**
  Specifies value of the custom field for the folder. A custom field can be regarded as a resource property and is available when it is enabled.

Automatically Convert Old Report Schema

Specifies whether or not JReport Server converts JReport reports of earlier versions into current version JReport reports when publishing the earlier version reports.

Apply Archive Policy

Applies an archive policy to the resource versions.

- **Archive as New Version**
  Specifies whether to use multiple versions for the resources.
  - **Maximum Number of Versions**
    Specifies the maximum number of versions that will be listed in the version table of the resource. The default value is 0, which means that the version number is unlimited.
- **Replace Old Version**
  Specifies to replace the old version when a new version is generated.

Set Permissions

Available only when the resources are to be published to the Public Reports or Public Components folder. Click the link to set user permissions to the resources in the Set Permissions dialog.

Advanced Publish

Specifies to publish resources in an advanced way.

- **Checkbox**
  Specifies which resources in the zip file you want to publish.
- **File Name**
  Displays the file name of the resource.
- **Resource Name**
  Specifies the name of the resource. This name is required and is used as the display name of the resource in the server resource tree.
- **Description**
  Specifies the description of the resource.
- **[Custom Field Name]**
  Specifies value of the custom field for the resource.
- **Advanced**
  Specifies the advanced properties for the resource, which include the archive policy of the resource, and the user permissions on the resource if the resource is to be published to the Public Reports or Public Components folder.

OK

Retains the settings and submits the task to server.

Cancel

Cancels operations.

Help

Displays the help document about this feature.

Related topics:

- Publishing resources

Report Properties dialog

You can set the properties of a specified report in its Properties dialog according to your requirements.

The Properties dialog is divided into the following two tabs:

- General
- Permission

Reset

Discards your modifications and restores the dialog to its default status.

OK

Retains the settings and submits the task to server.

Cancel

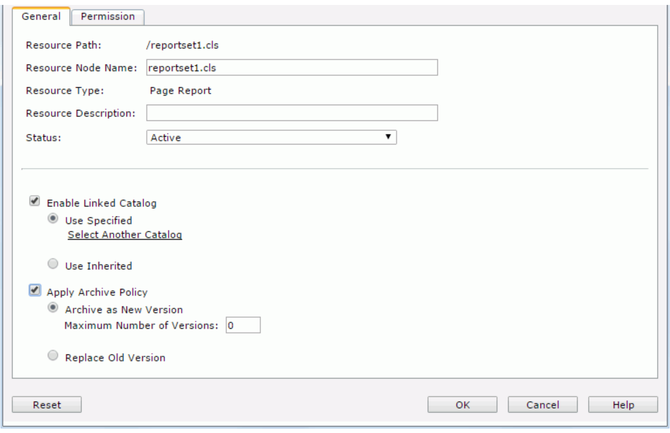
Cancels any settings and closes the dialog.

Help

Displays the help document about this feature.

General tab

Specifies the general properties of the report.



Resource Path

Shows the resource path.

Resource Node Name

Specifies the name for the report.

Resource Type

Shows the type of the resource.

Resource Description

Specifies the description for the report.

Status

Specifies the status of the report. If not specified, the status will be Active by default.

- **Active**
  The report can be run, advanced run and scheduled.
- **Inactive**
  The report cannot be run, advanced run or scheduled.
- **Incomplete**
  The report is not completely designed and cannot be run, advanced run or scheduled.

National Language Support

Specifies whether to enable the NLS feature for the report. This option is available when the Properties dialog is opened from the JReport Administration page.

[Custom Field Name]

Specifies value of the custom field for the report. A custom field can be regarded as a resource property and is available when it is enabled.

Choose Profile

Specifies the Page Report Studio profile to be applied to run the report which contains a set of Page Report Studio settings. This option is available when the Properties dialog is opened from the JReport Administration page.

Enable Linked Catalog

Enables to link the report with a catalog. If you have linked a report with a catalog, then even if the report and the catalog are not in the same directory, it also can be run with the catalog.

When you background run, advanced run or schedule the report, the linked catalog is applied instead of the catalog that is resided in the parent folder and originally used to run the report. For Advanced Run and Schedule, you can change the catalog to apply another one using the Select Another Catalog option in the General tab of the corresponding dialog.

- **Use Specified**
  Links the report with a catalog in the server resource tree.
  - **Select Another Catalog**
    Specifies another catalog in the Select Another Catalog dialog.
- **Use Inherited**
  Links the report with the linked catalog inherited from its parent folder. Note that if the parent folder does not enable linked catalog, you are not allowed to check this option.

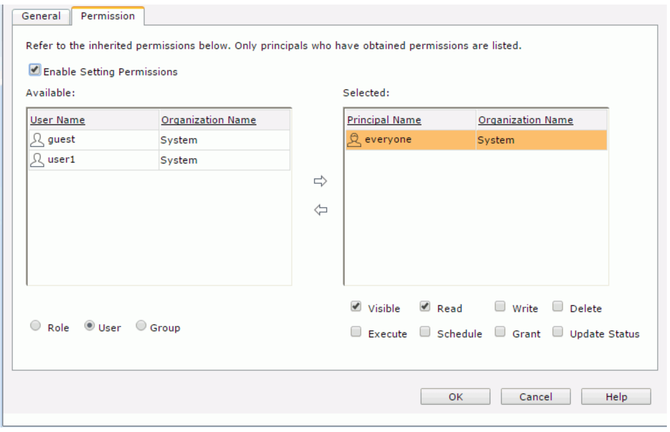
Apply Archive Policy

Applies an archive policy to the report versions.

- **Archive as New Version**
  Specifies whether to use multiple versions for the report.
  - **Maximum Number of Versions**
    Specifies the maximum number of versions that will be listed in the version table of the report. The default value is 0, which means that the version number is unlimited.
- **Replace Old Version**
  Specifies to replace the old version when a new version is generated.

Permission tab

Specifies permissions of roles/users/groups on the report. This tab is available when the report is in the Public Reports folder.



Enable Setting Permissions

Enables the setting of permissions.

Available

Lists the roles/users/groups to which you can assign permissions. When organization is enabled, the Organization Name column will be available in the table.

- **Role**
  If checked, all roles will be displayed in the Available box for you to assign permissions.
- **User**
  If checked, all users will be displayed in the Available box for you to assign permissions.
- **Group**
  If checked, all groups will be displayed in the Available box for you to assign permissions.

Adds the selected role, user or group to the Selected box.

Removes the selected role, user or group from the Selected box.

Selected

Select a role/user/group in the Selected box and then select the permissions you would like the role/user/group to have on the report. For what each permission is used for, see Permission.

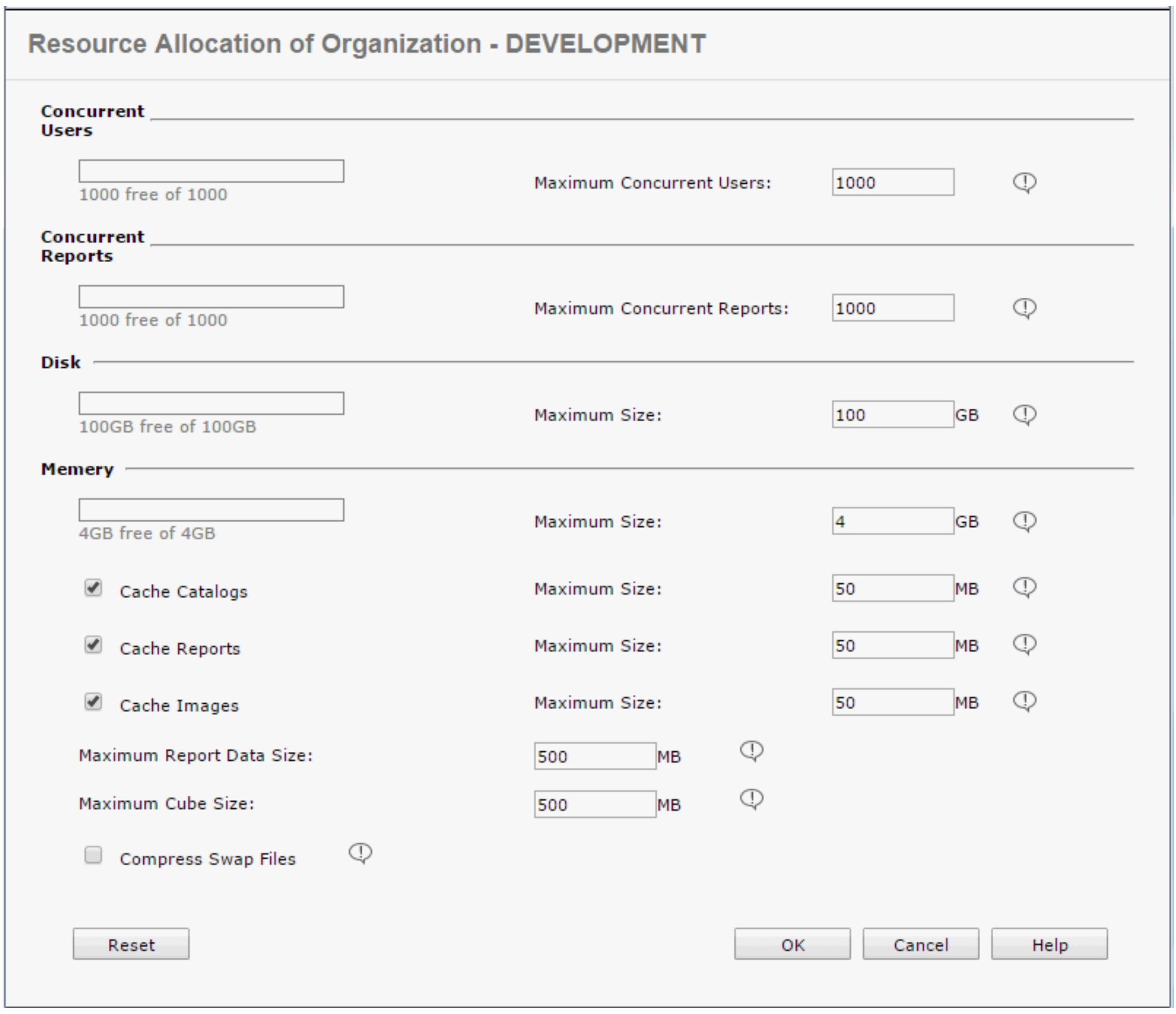
When organization is enabled, the Organization Name column will be available in the table.

Related topics:

- Changing resource and folder properties

Resource Allocation dialog

The dialog is displayed when you click the Resource Allocation link in the Control column of the organization table on the Organizations panel. It helps you to allocate the server resources to a specific organization.



Concurrent Users

Shows the number of concurrent users available.

- **Maximum Concurrent Users**
  Specifies the maximum number of concurrent users allowed in the organization. The concurrent organization users at runtime is under the control of the maximum number of concurrent users permitted by the server license.

Concurrent Reports

Shows the number of concurrent reports available.

- **Maximum Concurrent Reports**
  Specifies the maximum number of concurrent reports allowed in the organization. The concurrent organization reports at runtime is under the control of the maximum number of concurrent reports permitted by the server license.

Disk

Shows the disk size available.

- **Maximum Size**
  Specifies the maximum disk size that the organization can use, including disk space for resources and result files (the server history folder), log files (the server logs folder; in a cluster, log size is the sum of the log size in all the nodes), cached data and cube files (the server cache folder), even some temporary runtime files that are created when the organization users run reports/dashboards/Visual Analysis (the server temp folder).

Memory

Shows the memory size available.

- **Maximum Size**
  Specifies the maximum memory size that the organization can use.
- **Cache Catalogs**
  Specifies whether to cache catalogs used by the organization.
  - **Maximum Size**
    Specifies the maximum memory size that can be used for caching catalogs. The value should be no more than 80% of the maximum memory size.
- **Cache Reports**
  Specifies whether to cache reports used by the organization.
  - **Maximum Size**
    Specifies the maximum memory size that can be used for caching reports. The value should be no more than 80% of the maximum memory size.
- **Cache Images**
  Specifies whether to cache images used by the organization.
  - **Maximum Size**
    Specifies the maximum memory size that can be used for caching images. The value should be no more than 80% of the maximum memory size.
- **Maximum Report Data Size**
  Specifies the maximum memory size that can be used for caching report data within the organization. The value should be no more than 80% of the maximum memory size.
- **Maximum Cube Size**
  Specifies the maximum memory size for the cube used by the organization. The value should be no more than 80% of the maximum memory size.
- **Compress Swap Files**
  Specifies whether or not to compress the temporary data generated during runtime before it is swapped to disk. By compressing the swap files, the I/O efforts in certain circumstance may be remarkably reduced so that the overall performance can be improved.

  Note: Compressing swap files will increase CPU pressure because it uses compress algorithm to shrink data, so if your system already has high CPU usage, enabling this option will bring extra performance impact, depending on different circumstance, and such impact may overcome the performance gain that comes from reducing I/O time.

Reset

Resets to the default settings.

OK

Applies the selected values to the parameter and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Help

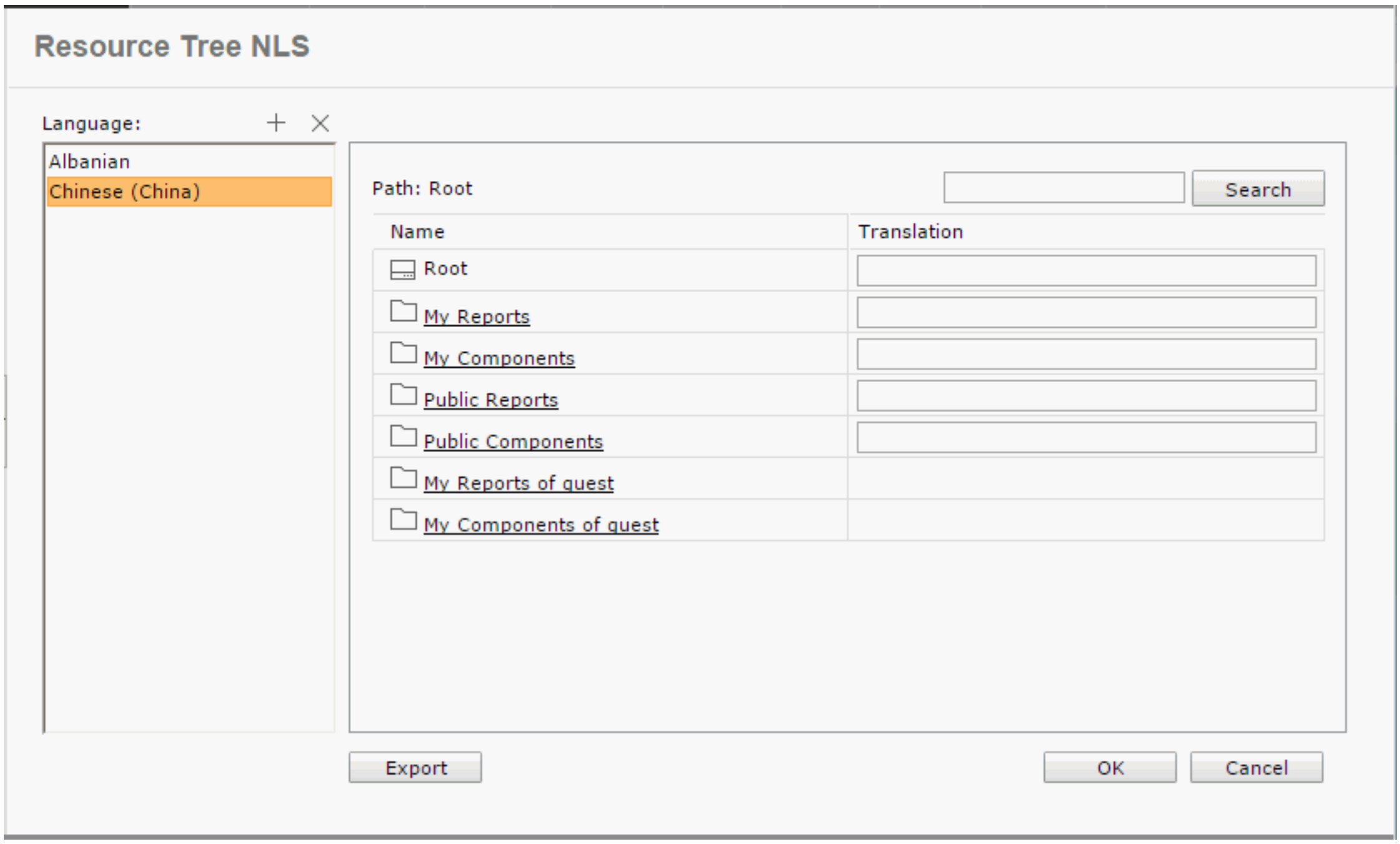
Displays the help document about this feature.

Related topics:

- Managing organizations

Resource Tree NLS dialog

This dialog appears when you click Resources > Resource Tree NLS on the JReport Administration page. It helps you to edit NLS for resources in the resource tree.



Language

Specifies the languages into which the resources will be translated.

Opens the Select Language Source dialog to choose where to add a language.

Removes the selected language from the Language box.

Path

Shows current path with links to parent folders.

Search

Searches for the Name and Translation items that match the search condition in the Search text box. Here you are not allowed to search wildcards. The results will be listed in the table below.

Name

Lists the resource names in the original language.

Translation

Specifies to which the resource names will be translated in the target language.

Export

Specifies to export the resource tree NLS map to a resource tree NLS file in the specified location for further use.

Note that only the resource tree of one translated language can be exported at a time and a user can only use the resource tree NLS file exported by himself.

OK

Applies the settings and closes the dialog.

Cancel

Cancels the settings and exits the dialog.

Help

Displays the help document about this feature.

Related topics:

- Editing resource tree NLS

Result Properties dialog

You can set the properties of a specified result in its Properties dialog according to your requirements.

The Properties dialog is divided into the following two tabs:

- General
- Permission

Reset

Discards your modifications and restores the dialog to its default status.

OK

Retains the settings and submits the task to server.

Cancel

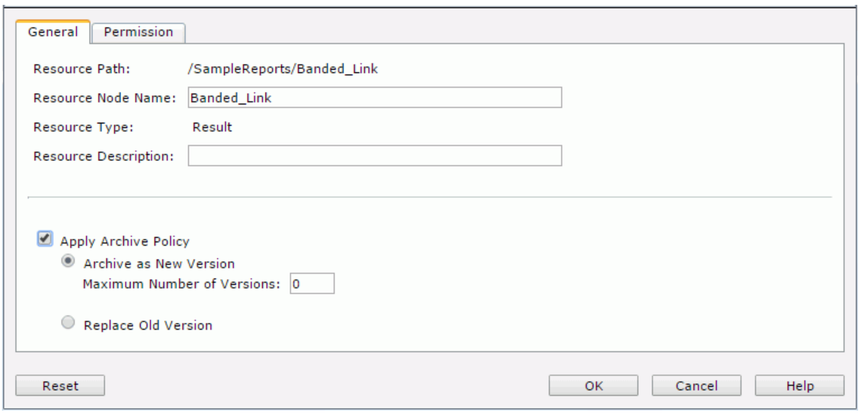
Cancels any settings and closes the dialog.

Help

Displays the help document about this feature.

General tab

Specifies the general properties of the result.



Resource Path

Shows the resource path.

Resource Node Name

Specifies the name for the result.

Resource Type

Shows the type of the resource.

Resource Description

Specifies the description of the result.

[Custom Field Name]

Specifies value of the custom field for the result. A custom field can be regarded as a resource property and is available when it is enabled.

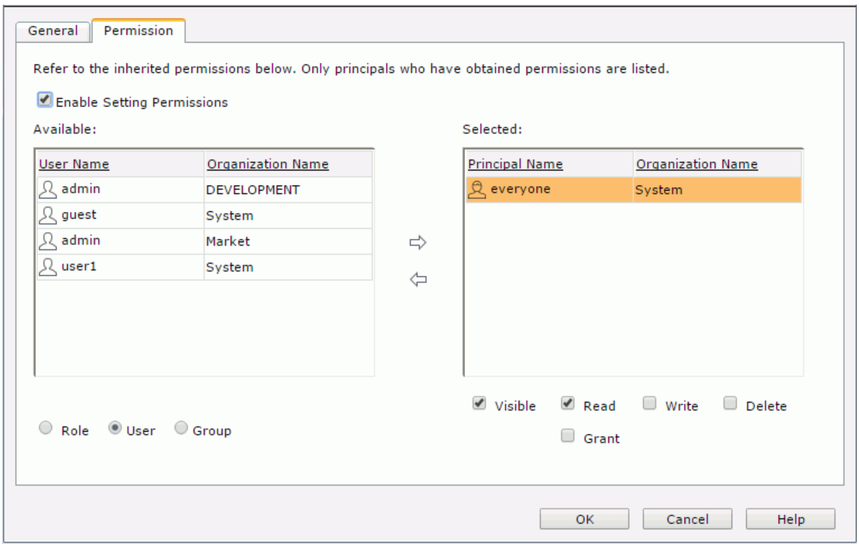
Apply Archive Policy

Applies an archive policy to the result versions.

- **Archive as New Version**
  Specifies whether to use multiple versions for the result.
  - **Maximum Number of Versions**
    Specifies the maximum number of versions that will be listed in the version table of the result. The default value is 0, which means that the version number is unlimited.
- **Replace Old Version**
  Specifies to replace the old version when a new version is generated.

Permission tab

Specifies permissions of roles/users/groups on the result. This tab is available when the result is in the Public Reports folder.



Enable Setting Permissions

Enables the setting of permissions.

Available

Lists the roles/users/groups to which you can assign permissions. When organization is enabled, the Organization Name column will be available in the table.

- **Role**
  If checked, all roles will be displayed in the Available box for you to assign permissions.
- **User**
  If checked, all users will be displayed in the Available box for you to assign permissions.
- **Group**
  If checked, all groups will be displayed in the Available box for you to assign permissions.

Adds the selected role, user or group to the Selected box.

Removes the selected role, user or group from the Selected box.

Selected

Select a role/user/group in the Selected box and then select the permissions you would like the role/user/group to have on the result. For what each permission is used for, see Permission.

When organization is enabled, the Organization Name column will be available in the table.

Related topics:

- Changing resource and folder properties

Schedule dialog

JReport Server allows you to schedule a report to run at a specific time and publish the report result to different formats.

To access the Schedule dialog, on the JReport Console > Resources page, browse to the report you want to schedule, then do either of the following:

- Select the row the report is in, then on the task bar of the Resources page, click Run > Schedule.
- Select the report row, right-click in the row and select Schedule from the shortcut menu.
- Put the mouse pointer over the report row and click the Schedule button on the floating toolbar.

This dialog contains six tabs as listed below. Set the settings in each tab according to your requirements.

- General tab
- Parameter tab
- Publish tab
- Conditions tab
- Notification tab
- Duration tab

Back

Goes back to the left tab.

Next

Goes to the right tab.

Finish

Applies the settings.

Cancel

Closes the dialog and discards any changes.

Help

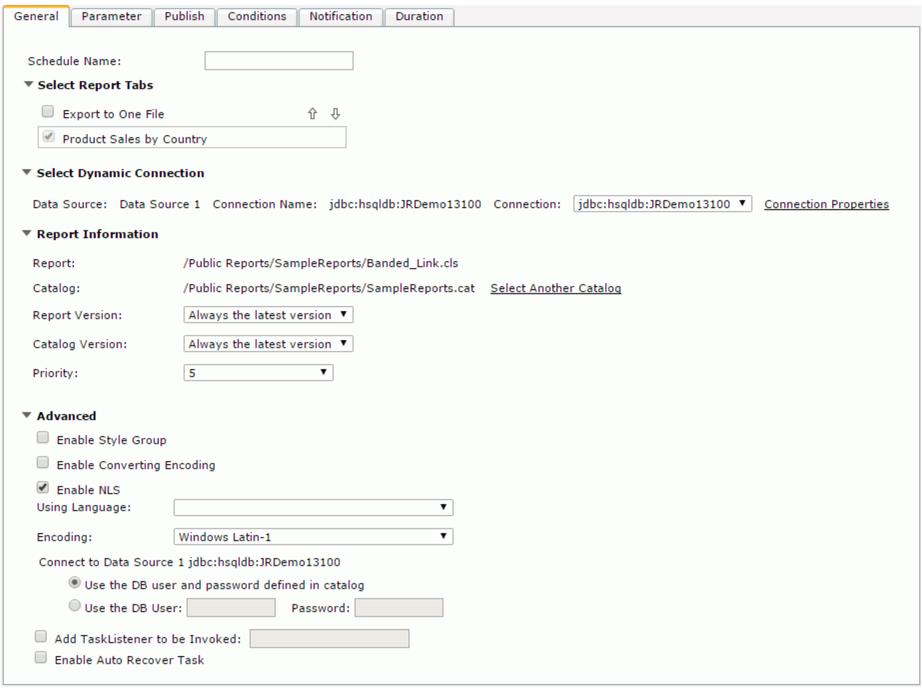
Displays the help document about this feature.

Related topics:

- Scheduling reports with dialog

General tab

Before you can schedule a task, you first need to configure the settings in this tab.



Schedule Name

Specifies the name for the schedule task.

Select Report Tabs

The section is available for page reports only. It allows you to specify the report tabs you schedule to run. The selected report tabs will be exported in the list order. If the page report has only one report tab, it is selected by default.

Export to One File

If checked, the selected report tabs will be exported to one result file.

⇧

Moves the specified report tab one step up.

⇩

Moves the specified report tab one step down.

Select Dynamic Connection

The section is available when there are multiple dynamic connections for the current log-in user. It allows you to specify a dynamic connection.

Data Source

Displays the data source name in the catalog.

Connection Name

Displays the dynamic connection name.

Connection

Specifies a dynamic connection from the drop-down list.

Connection Properties

Shows the information of the selected dynamic connection, which is read only. For details about the propeties, refer to Edit Dynamic Connection dialog.

Report Information

Specifies the report information.

Report

Lists information about the report.

Catalog

Lists the catalog information.

Select Another Catalog

Specifies another catalog for the report in the Select Another Catalog dialog.

Report Version

Specifies the report version used for the schedule task. The default version is always the latest version.

Catalog Version

Specifies the catalog version used for the schedule task. The default version is always the latest version.

Priority

Specifies the priority level of the schedule task. The priority levels are from 1 to 10 in ascending order of lowest priority to highest priority. This property is available to administrators. By default this property is ignored unless server.properties is modified to set queue.policy not equal to 0.

Advanced

Configures some advanced settings.

Enable Style Group

If a style group has been specified from the Override Style Group drop-down list in the Profile > Customize Server Preferences > Advanced tab, this option gets checked by default and the specified default style group is selected in the Style Group drop-down list, which will be applied to run all the selected report tabs or web report for all export formats. You can also choose another style group from the Style Group drop-down list to run the page report tabs or web report.

When the <No Style> item is selected in the Override Style Group drop-down list, this option is unchecked by default and the style group properties of the selected report tabs or web report for the corresponding export formats that are predefined in JReport Designer will be used in the schedule task. These predefined style group properties take effect when publishing to version, disk, e-mail and FTP.

Enable Converting Encoding

Specifies to enable the conversion of encoding. From the Before Converting and After Converting drop-down lists, select encodings as required.

Enable NLS

Specifies whether to enable NLS for the report. If checked, the Using Language drop-down list will be displayed for you to choose a language. If there is no NLS resource defined for the report, you can only run the report using the default language.

Encoding

Specifies the encoding of the report from the drop-down list.

Connect to [Data Source Name]

Specifies the DB user and password with which you want to connect to the data source the report uses.

- **Use the DB user and password defined in catalog**
  If checked, the DB user and password defined in the catalog will be used.
- **Use the DB User**
  If checked, specify another DB user and password instead of the one defined in the catalog.

Add TaskListener to be Invoked

Enables you to call the Java application before/after viewing or when scheduling the report.

Specify a preferred server to run the task

Enables you to directly specify a server in a cluster to perform the schedule task instead of using load balancing. This option is available only when there are more than one active server in a cluster and when the Identify Server Preference option in the Profile > Customize Server Preferences > Advanced tab is selected.

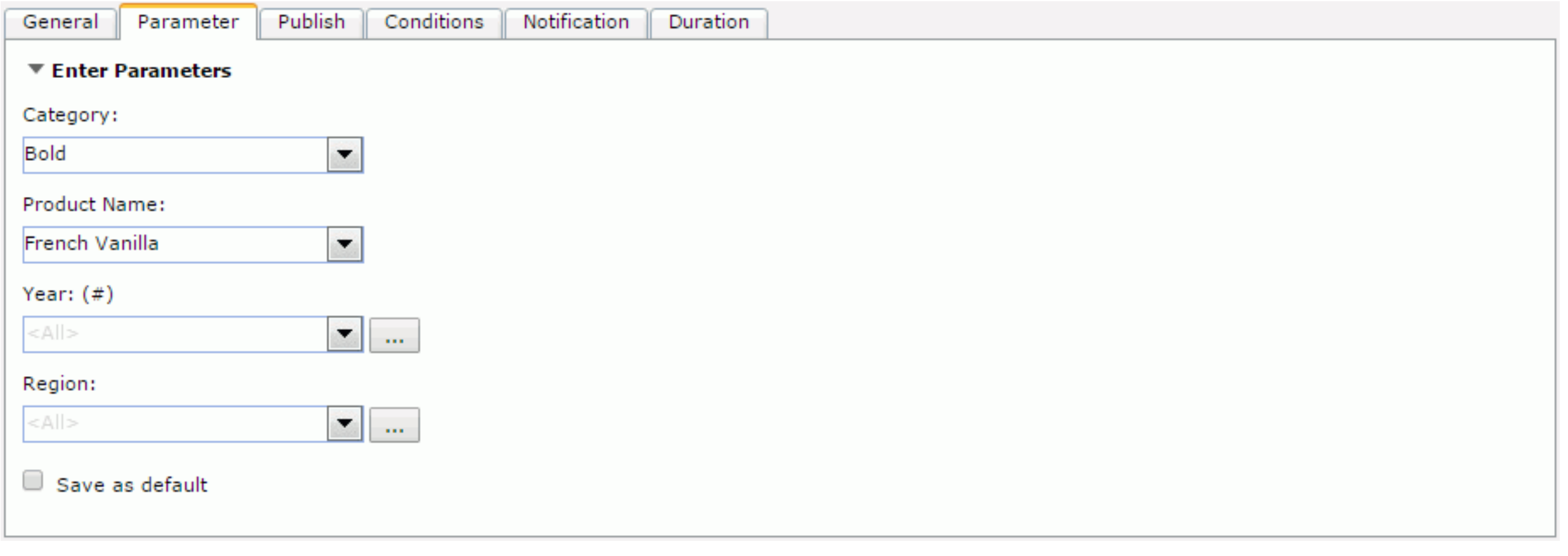
Enable Auto Recover Task

Specifies to enable the task to be auto recovered.

- **Maximum Retry Times**
  Specifies the maximum number of times in which to retry running the task in order to recover it.
- **Retry Interval**
  Specifies the interval between retries.
- **Recreate All Results**
  Specifies whether to recreate all or just the failed results when you published the task to multiple formats.

Parameter tab

This tab allows you to specify the parameter values if any to run the report.



Enter Parameters

Displays parameters of the report with your last-time saved default values, which could be the values saved in this tab last time, or in the Enter Parameter Values dialog or Parameter Settings dialog, or when advanced running the report, and for a web report, could also be the values saved in the Parameters panel. Edit the values according to your requirement. You may specify parameter values in these ways.

If you have not yet set the default values on the server, or if you did but your saved default values cannot fully match the current parameters, all the parameters will use their default values specified in the parameters' definition as the initial values.

If no parameter is used in the report, "No Parameter Needed" will be displayed here.

Use Saved Values

If it is available, you can select the previously saved parameter values to apply to the report and save parameter values for reuse later.

Save as default

Saves current parameter values as the default values for the report. It is a user-report level setting.

This option is an action and it takes effect after the task is submitted. Its initial status is always unselected.

For page reports, this option is available when Enable Setting Default Parameter Values For Page Report is checked in the Profile > Customize Server Preferences > Advanced tab.

For web reports, this option is available when Enable Setting Default Parameter Values For Web Report is checked in the Profile > Customize Server Preferences > Advanced tab.

Publish tab

The Publish tab allows you to specify the type of the task. There are two main tasks to publish your reports: Default Task and User Task. By default, Default task is used to publish the reports. To use the User Task, click the User Task link in the Default Task page.

Default Task

If you select Default Task to publish your reports with, when specifying your task type, you can choose from the following six sub tasks:

- To Version
  Publishes the report result to the versioning system.
- To Disk
  Publishes the report result to the file system. This option is not available to organization users when the organization feature is enabled.
- To E-mail
  Publishes the report result to e-mail.
- To Printer
  Publishes the report result to a printer.
- To Fax
  Publishes the report result to fax.
- To FTP
  Publishes the report result to an FTP site.

Note: If you are scheduling a bursting report, the Publish tab will change according to the bursting setting made in the General tab of the Schedule dialog. For details about the change, see Scheduling a task containing a bursting report.

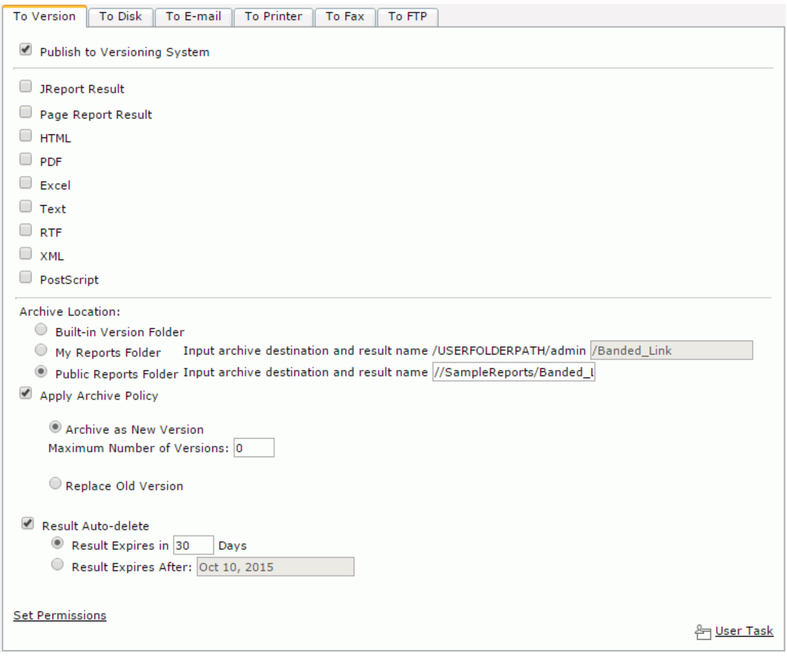
User Task

If you select User Task to publish your reports, you can implement a customized task with schedule properties. This option is not available to organization users when the organization feature is enabled.

- **User Task Class Name**
  Specifies the class name for the user task.
- **Display Name**
  Specifies the display name for the user task.
- **User Task Properties**
  Specifies user task properties to define tasks.
- **Import User Task Properties from File**
  Imports a user task from a properties file.
- **Upload Properties**
  Uploads the properties to JReport Server. The properties will be listed in the User Task Properties area.

To Version

This tab is for you to specify settings for publishing the report result to the versioning system.



Publish to Versioning System

Specifies to publish the report result to the versioning system. You can publish the result to the following formats:

- **JReport Result**
  Publishes the report to a JReport result file.
  - **Zip Result**
    Specifies to compress the result to reduce the disk size and I/O; however, it uses more CPU resources.
  - **Run Linked Report**
    If the report is linked with other reports, you can check whether or not to generate the linked reports in the JReport result file. If you are only interested in the primary report, leave this option unchecked. Generating linked reports at the same time, especially when the linked reports contain a large amount of data, will cause performance issue.
  - **Precision Level**
    Specifies the precision level with which to publish the report result. Note that changing the default value may cause abnormalities in report layout.
- **Page Report Result**
  Publishes the report to a page report result file (RSD file) that can be viewed in a Page Report Studio window. Some reports cannot be published to version in the Page Report Result format, such as self-contained (CLX) reports, multi-level nested master and subreports, reports containing data objects, and reports developed from a dynamic query or hierarchical data source (HDS).

  An RSD file is a middle result for page report that can preserve data and the working status of a page report and can be shared with others. Based on the RSD file, you can rebuild a page report to analyze data and to export to other formats. An RSD file is a standalone working file so all the useful information is added into it, such as catalog, report template, NLS, data, parameters and security. The security information of an RSD file is the user of the generator that creates it. When different users open the same RSD, they will get the same result based on the same security configuration. You cannot refresh a report viewed from an RSD file to refetch data from the database. When the system rebuilds a report according to the RSD file, it will only load data from the RSD file. When opening a report RSD, you cannot go back to the actions which the RSD generator took before generating the RSD file but can take actions based on the current RSD resources.
  - **Zip Result**
    Specifies to compress the result. The compressed file's suffix will still be .rsd, it's size will be smaller which reduces IO and disk usage; however, it uses more CPU resources.
  - **Precision Level**
    Specifies the precision level with which to publish the report result. Note that changing the default value may cause abnormalities in report layout.
- **Web Report Result**
  Publishes the web report to a static web report result file (WST file). WST files can be exported to HTML, PDF, Text, Excel, XML, RTF, and Postscript formats.
  - **Zip Result**
    Specifies to compress the result to reduce the disk size and I/O; however, it uses more CPU resources.
  - **Run Linked Report**
    If the report is linked with other reports, you can check whether or not to generate the linked reports in the WST file. If you are only interested in the primary report, leave this option unchecked. Generating linked reports at the same time, especially when the linked reports contain a large amount of data, will cause performance issue.
  - **Precision Level**
    Specifies the precision level with which to publish the web report result. Note that changing the default value may cause abnormalities in report layout.
- HTML
  Publishes the report result to the versioning system in HTML format.
- PDF
  Publishes the report result to the versioning system in PDF format.
- Excel
  Publishes the report result to the versioning system in Excel format.
- Text
  Publishes the report result to the versioning system in Text format.
- RTF
  Publishes the report result to the versioning system in RTF format.
- XML
  Publishes the report result to the versioning system in XML format.
- PostScript
  Publishes the report result to the versioning system in PostScript format.

Archive Location

Specifies the location for the saved report result version.

- **Built-in Version Folder**
  Specifies to save the report result version to the built-in version folder. When the organization feature is enabled and the report is from the Public Reports folder, this option is not available to organization users.
- **My Reports Folder**
  Specifies to save the report result version to the My Reports folder.
- **Public Reports Folder**
  Specifies to save the report result version to the Public Reports folder. Not available to organization users.
- **Organization Reports Folder**
  Specifies to save the report result version to the Organization Reports folder. Available to organization users only.

Apply Archive Policy

Applies an archive policy to the report result version.

- **Archive as New Version**
  Specifies whether to use multiple versions for the report result.
  - **Maximum Number of Versions**
    Specifies the maximum number of versions that will be listed in the version table of the report result. The default value is 0, which means that the version number is unlimited.
- **Replace Old Version**
  Specifies to replace the old version when a new version is generated.

Result Auto-delete

Specifies the deleted date of the result. Set the result to be automatically deleted within one hundred years. If the time you specify exceeds one hundred years, JReport Server will keep the report result forever.

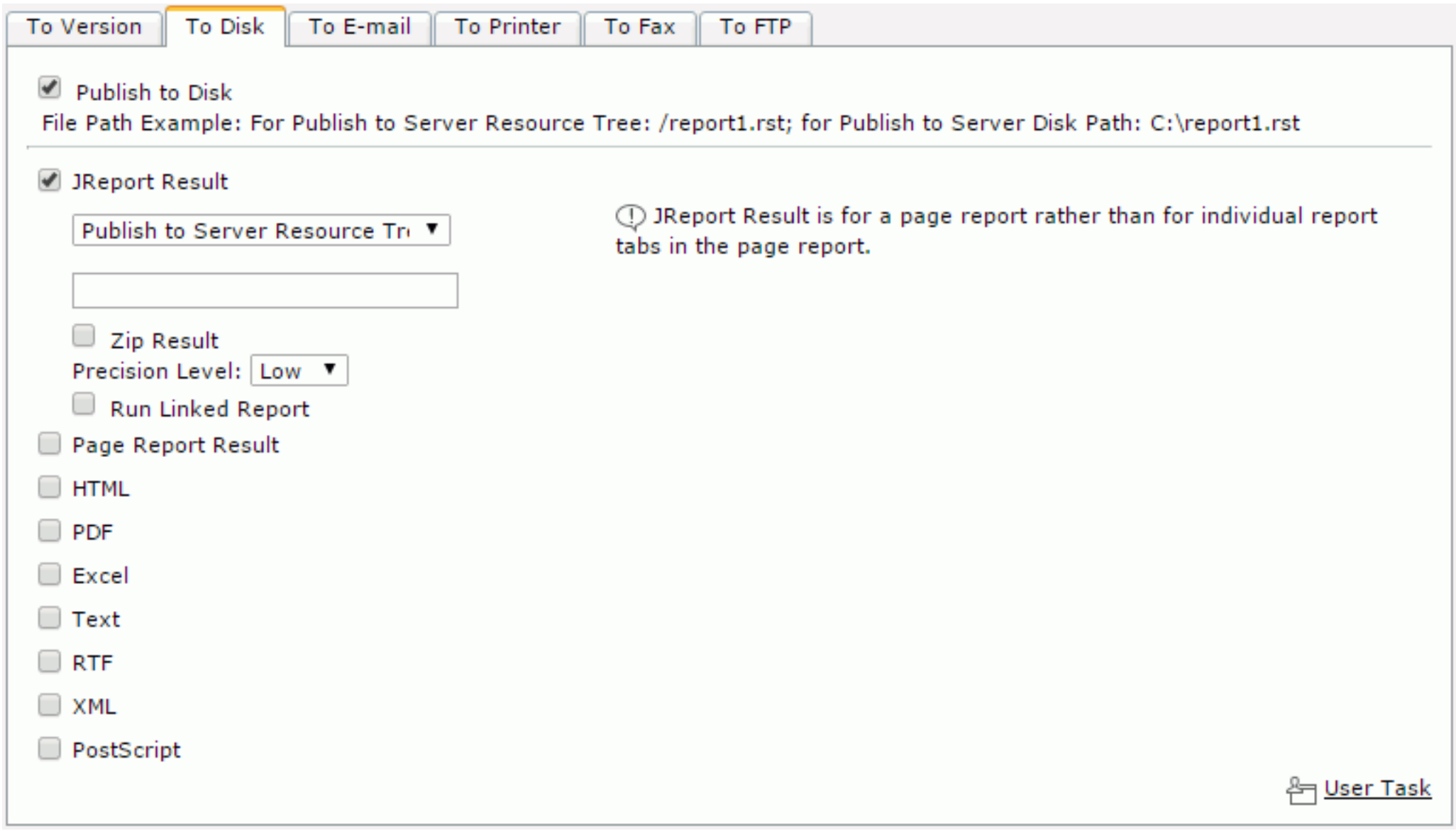
- **Result Expires in N Days**
  Specifies a period after which the report result will be automatically deleted.
- **Result Expires After**
  Specifies a certain day on which the report result will be automatically deleted.

Set Permissions

Available only when the Archive Location is specified to be Public Reports Folder. Click the link to set user permissions to the specified report tabs or web report in the Set Permissions dialog.

To Disk

This tab is for you to specify settings for publishing the report result to the file system.



Publish to Disk

Specifies to publish the report result to the file system.

You can publish the report result to the following formats. For each format, you can decide whether to publish the results to the server resource tree or to the server disk path. Then you need to type in the blank location field root of the resource tree or disk path as shown in the examples below:

Example for server resource tree: `/report1.rst`

Example for server disk path: `C:\temp\report1.rst`

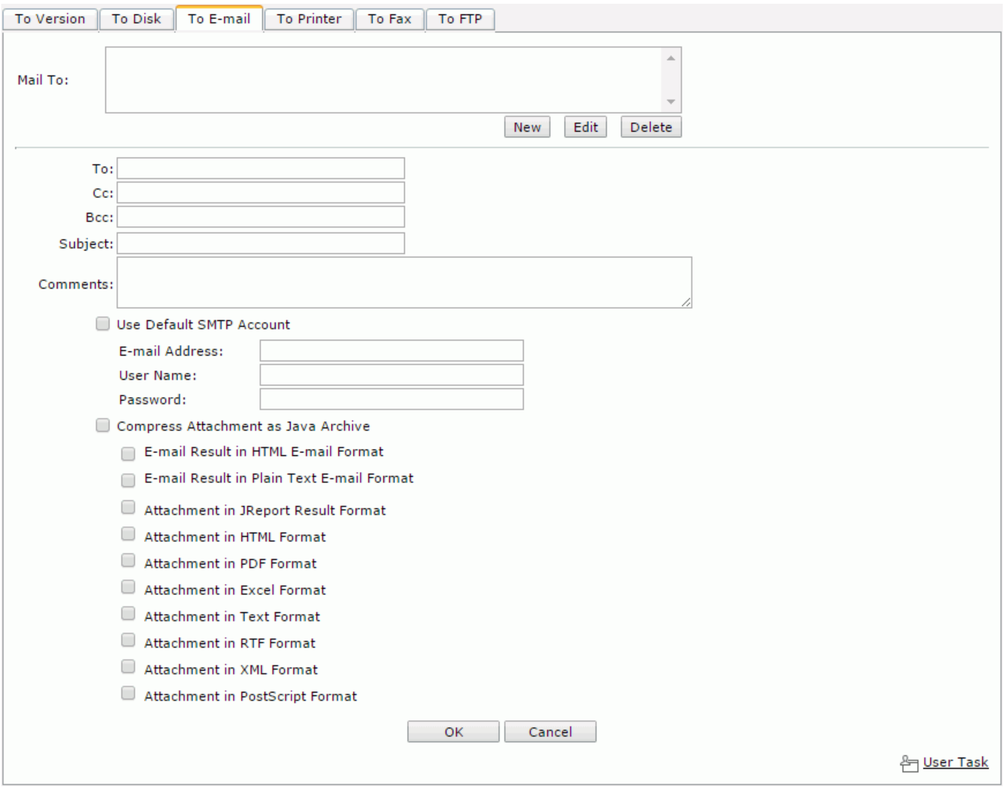
- JReport Result
  Publishes the report to a JReport result file (RST file).
- Page Report Result
  Publishes the report to a page report result file (RSD file) that can be viewed in a Page Report Studio window.
- Web Report Result
  Publishes the web report to a static web report result file (WST file).
- HTML
  Publishes the report result to the file system in HTML format.
- PDF
  Publishes the report result to the file system in PDF format.
- Excel
  Publishes the report result to the file system in Excel format.
- Text
  Publishes the report result to the file system in Text format.
- RTF
  Publishes the report result to the file system in RTF format.
- XML
  Publishes the report result to the file system in XML format.
- PostScript
  Publishes the report result to the file system in PostScript format.

Notes:

- For the JReport Result and Page Report Result formats, you need only provide one file path because all selected report tabs will be output into one file. For other formats, if you leave the Export to One File option in the General tab unchecked, you need to specify a file path for each selected report tab.
- The files with the extension of .rsd should be opened with Page Report Studio and those with the extension of .rst should be opened by JReport Server.

To E-mail

This tab is for you to specify settings for publishing the report result to e-mail.



Mail To

Lists the e-mail addresses you have sent mail to.

New

Creates a new e-mail.

- **To**
  Specifies the address you want to send the e-mail to.
- **Cc**
  Specifies the address you want to copy to.
- **Bcc**
  Specifies the address you want to secretly copy to.
- **Subject**
  Specifies the subject of the e-mail.
- **Comments**
  Specifies the contents of the e-mail or comments to the contents.
- **Use Default SMTP Account**
  Specifies whether to use the default SMTP account configured on the JReport Administration page > Configuration > E-mail Server panel as the e-mail publish account.

  The option is checked by default. Uncheck it and you will be able to specify a different account using the following options:

  - **E-mail Address**
    Specifies the e-mail address of the new SMTP account.
  - **User Name**
    Specifies the user name of the new account.
  - **Password**
    Specifies the password of the new account.
- **Compress Attachment as Java Archive**
  Specifies whether to compress the mail attachment as Java Archive.
- **E-mail Result in HTML E-mail Format**
  Sends the report result via e-mail to the specified address in HTML format. The report result will be shown in HTML format in the mail body.
  - **Run Linked Report**
    If the report is linked with other reports, you can check whether or not to generate the linked reports in the HTML result. If you are only interested in the primary report, leave this option unchecked. Generating linked reports at the same time, especially when the linked reports contain a large amount of data, will cause performance issue.
- **E-mail Result in Plain Text E-mail Format**
  Sends the report result via e-mail to the specified address in plain text format. The report result will be shown in plain text format in the mail body with no other information such as color, images and so on.

  Notes:

  - The two formats, E-mail Result in HTML E-mail Format and E-mail Result in Plain Text Format, cannot be support concurrently.
  - If you choose to send the mail in E-mail Result in HTML E-mail Format, the comments that you input for the mail will be overwritten by the report result.

When you select one of the following formats, that is you want to send the report result as an attachment file, you can specify a file name for the file by setting the File Name option.

- **Attachment in JReport Result Format**
  Sends the report result via e-mail to the specified address with a JReport result file as attachment.
- **Attachment in Web Report Result Format**
  Sends the web report result via e-mail to the specified address with a WST file as attachment.
- **Attachment in HTML Format**
  Sends the report result via e-mail to the specified address with an HTML file as attachment.
- **Attachment in PDF Format**
  Sends the report result via e-mail to the specified address with a PDF file as attachment.

  Note: If the Split PDF option is specified in the Configuration > Export > E-mail tab on the JReport Administration page, the TOC and Sign options are disabled here.
- **Attachment in Excel Format**
  Sends the report result via e-mail to the specified address with an Excel file as attachment.
- **Attachment in Text Format**
  Sends the report result via e-mail to the specified address with a Text file as attachment.
- **Attachment in RTF Format**
  Sends the report result via e-mail to the specified address with a RTF file as attachment.
- **Attachment in XML Format**
  Sends the report result via e-mail to the specified address with an XML file as attachment.
- **Attachment in PostScript Format**
  Sends the report result via e-mail to the specified address with a PostScript file as attachment.
- **OK**
  Retains the settings and adds the e-mail address into the Mail To list.
- **Cancel**
  Cancels operations.

Edit

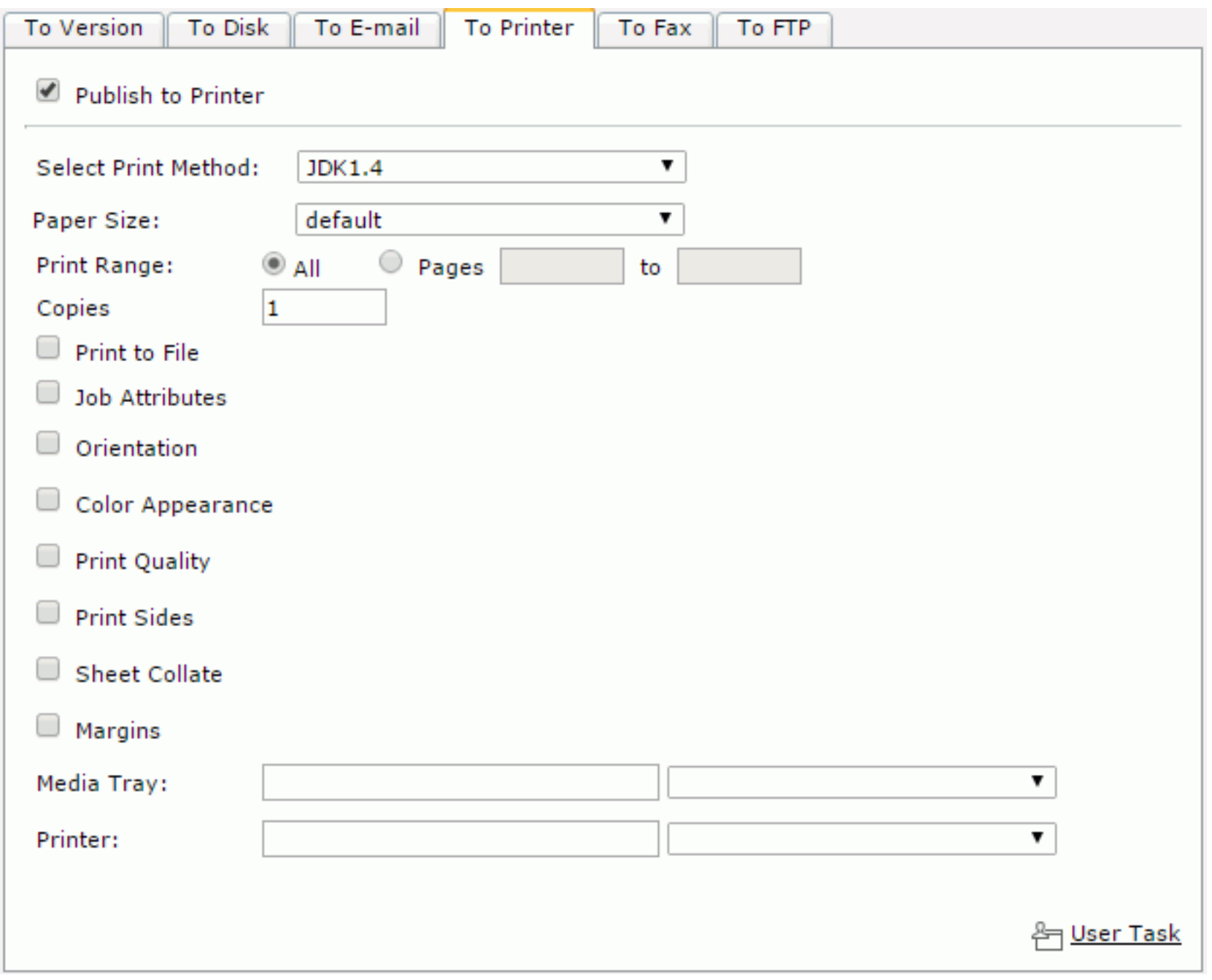
Edits the specified e-mail in the Mail To list.

Delete

Deletes the specified e-mail from the Mail To list.

To Printer

This tab is for you to specify settings for publishing the report result to a printer.



Publish to Printer

Specifies to publish the report result to a printer.

Select Print Method

Specifies the printing method to print the report result.

Printer

Specifies the name with the path of the printer. You can type it in the box or select it from the drop-down list.

When JDK1.4 is selected in the Select Print Method drop-down list, the following options will be available.

Paper Size

Specifies the paper size.

Print Range

Specifies the pages to be printed.

Copies

Specifies the number of copies you want to print. The number of copies will be applied to all specified pages.

Print to File

Prints the document to a file instead of a printer. This kind of files can only be opened and printed by serial port printer on Windows via DOS command.

- **File Name**
  Specifies the name of the file to which you print. If no path is provided in the file name, the file will be saved to the working directory of the server.

Job Attributes

Specifies the properties of the print task.

- **Priority**
  Specifies the print priority for this task.
- **Job Name**
  Specifies the name of the print task.
- **User Name**
  Specifies the name of the user.

Orientation

Specifies the orientation for the printed reports.

- **Landscape**
  Prints your reports in a Landscape orientation.
- **Portrait**
  Prints your reports in a Standard letter orientation.
- **Reverse Landscape**
  Prints your reports in a Reverse Landscape orientation.
- **Reverse Portrait**
  Prints your reports in a Reverse Portrait orientation.

Color Appearance

Specifies to print the reports in Monochrome or in Color.

Print Quality

Specifies the print quality. It can be Draft, Normal, or High.

Print Sides

Specifies the print sides for the printed reports.

- **One Side**
  Imposes each consecutive print-stream page upon the same side of consecutive media sheets.
- **Duplex**
  Imposes each consecutive pair of print-stream pages upon the front and back sides of consecutive media sheets, such that the orientation of each pair of print-stream pages on the medium is correct for the reader as if for binding on the long edge.
- **Tumble**
  Imposes each consecutive pair of print-stream pages upon front and back sides of consecutive media sheets, such that the orientation of each pair of print-stream pages on the medium is correct for the reader as if for binding on the short edge.

Sheet collate

Specifies the printing order for printing multiple-page reports.

- **Collated**
  If selected, when you print two copies of a three-page report, the page order is 1, 2, 3; 1, 2, 3.
- **Uncollated**
  If selected, when you print two copies of a three-page report, the page order is 1, 1; 2, 2; 3, 3.

Margins

Specifies the paper margins for the printed reports (MM or Inch).

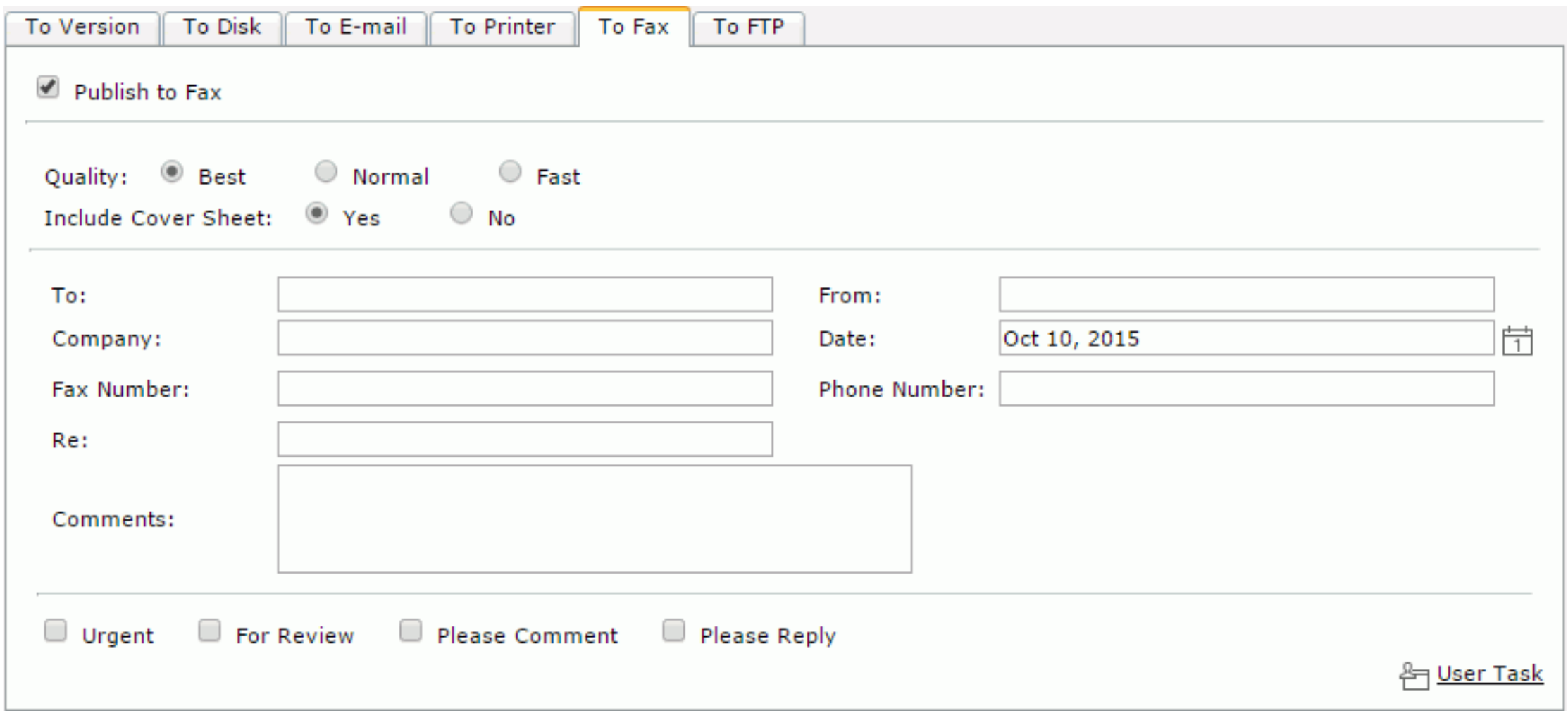
Media Tray

First select a printer, and then from the drop-down list, specify the media tray.

Note: JReport Server cannot obtain information on which options are supported and which are not from your printer and reflect it on this tab. You should configure the setting according to your printer carefully.

To Fax

This tab is for you to specify settings for publishing the report result to fax.



Note that before you can fax the report result, you must have your modem configured, otherwise a warning message will be displayed.

Publish to Fax

Specifies to publish the report result to fax.

Quality

Specifies the quality of the fax: Best, Normal, and Fast.

Include Cover Sheet

Specifies to send a cover sheet with the fax.

The following items are displayed on the fax cover sheet.

To

Specifies the name of the recipient.

From

Specifies the name of the sender.

Company

Specifies the sender's company.

Date

Specifies the date on which the fax is to be sent. You can specify the date by clicking the calendar button .

Fax Number

Specifies the fax number of the recipient.

Phone Number

Specifies the phone number of the sender.

Re

Specifies the subject of the fax.

Comments

Specifies the comments of the fax.

Urgent

Specifies whether the fax is urgent or not.

For Review

Specifies that the recipient only needs to view the fax.

Please Comment

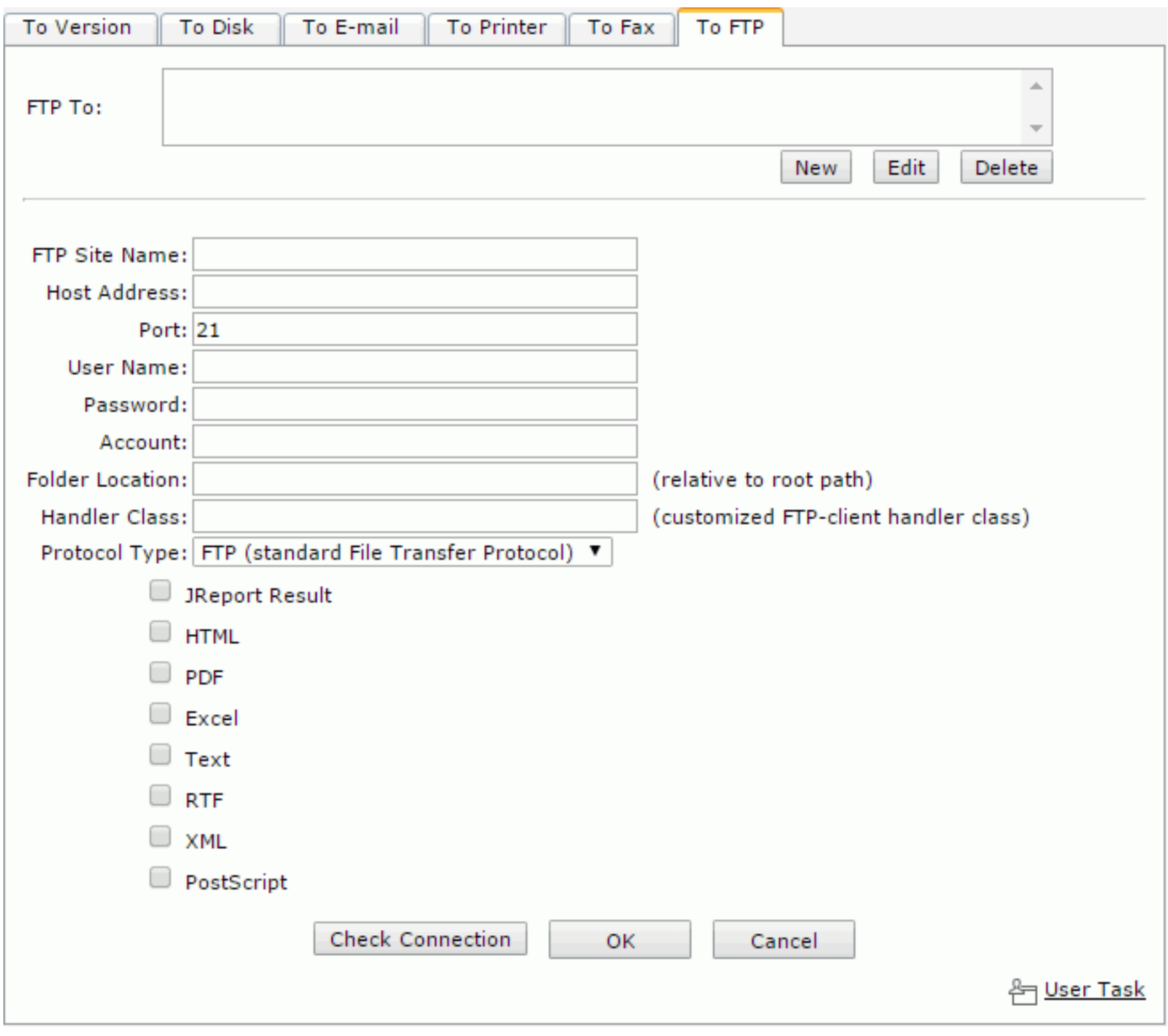
Specifies that the recipient is required to comment on the content of the fax.

Please Reply

Indicates that a reply is required for the fax.

To FTP

This tab is for you to specify settings for sending the report result to an FTP site.



FTP To

Lists the FTP sites you can send the report result to.

New

Creates a new FTP site.

- **FTP Site Name**
  A user-defined name of the FTP site. It is optional.
- **Host Address**
  The domain name or IP address of the FTP site. It cannot be null.
- **Port**
  The port of the FTP server. It is optional, and by default 21 is used for Standard FTP and Explicit FTPS, 22 SCP and SFTP, and 990 for Implicit FTPS.
- **User Name**
  The user name is valid to the authentication of the FTP server that can access the FTP site. If not specified, "anonymous" will be used as the user name by default.
- **Password**
  The password is valid to the authentication of the FTP server that enables the user name to access the FTP site.
- **Account**
  The account of the FTP user if there exists.
- **Folder Location**
  The location where to put the report result files on the FTP server. If not specified, the root path "/" of the FTP server will be used by default.
- **Handler Class**
  A customized FTP-client handler class can be used instead of the one provided in JReport. You should specify a fully-qualified class name that is package name plus class name, for example, test.DemoJakartaFTPHandler.
- **Protocol Type**
  Specifies the protocol type used for publising the report result to FTP.
  - **FTP (Standard File Transfer Protocol)**
    Files are transferred in a non-secure (plain text) style.
  - **SFTP using SSH2 (Secure Shell)**
    SFTP (Secure FTP) is a component of Secure Shell and is supported by most Unix servers running SSH2. Ask your server administrator about availability.
  - **SCP (Secure Copy)**
    This is a means of securely transferring computer files between a local and a remote host or between two remote hosts, using the Secure Shell (SSH) protocol. If this type is selected, you will not be able to create folder to the FTP server.
  - **FTP with TLS/SSL (Explicit)**
    Explicit connection type of FTP security with TLS/SSL.
  - **FTP with TLS/SSL (Implicit)**
    Implicit connection type of TP security with TLS/SSL.
- **Formats**
  Specifies in which formats to send the report result file to the FTP site. You can specify one or more formats. For each format, you need to specify a name for the result file.
  - JReport Result
    Sends the report result in a JReport result file to the specified FTP site.
  - Web Report Result
    Sends the web report result in a WST file to the specified FTP site.
  - HTML
    Sends the report result in an HTML file to the specified FTP site.
  - PDF
    Sends the report result in a PDF file to the specified FTP site.
  - Excel
    Sends the report result in an Excel file to the specified FTP site.
  - Text
    Sends the report result in a Text file to the specified FTP site.
  - RTF
    Sends the report result in a RTF file to the specified FTP site.
  - XML
    Sends the report result in an XML file to the specified FTP site.
  - PostScript
    Sends the report result in a PostScript file to the specified FTP site.
- **Check Connection**
  Checks whether to be able to connect to the specified FTP location with the specified user account.
- **OK**
  Retains the settings and adds the FTP site into the FTP To list.
- **Cancel**
  Cancels the creation of the new FTP site.

Edit

Edits the specified FTP site in the FTP To list.

Delete

Deletes the specified FTP site from the FTP To list.

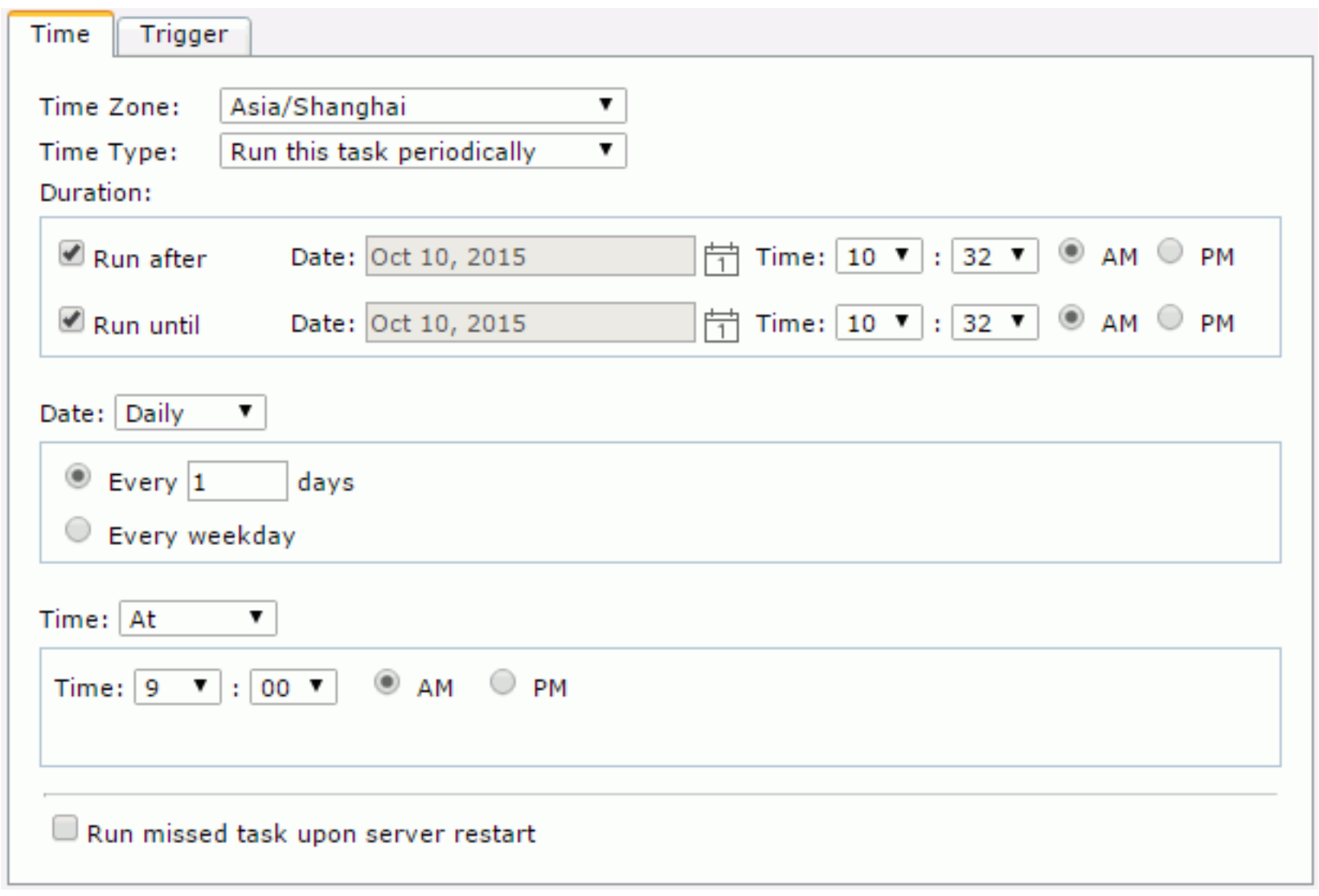
Conditions tab

The Conditions tab allows you to specify the conditions for the publishing tasks. It contains the following two sub tabs:

- Time tab
- Trigger tab

Time tab

The Time tab allows you to set settings for specifying the time for when a task is to be performed.



Time Zone

Specifies the time zone.

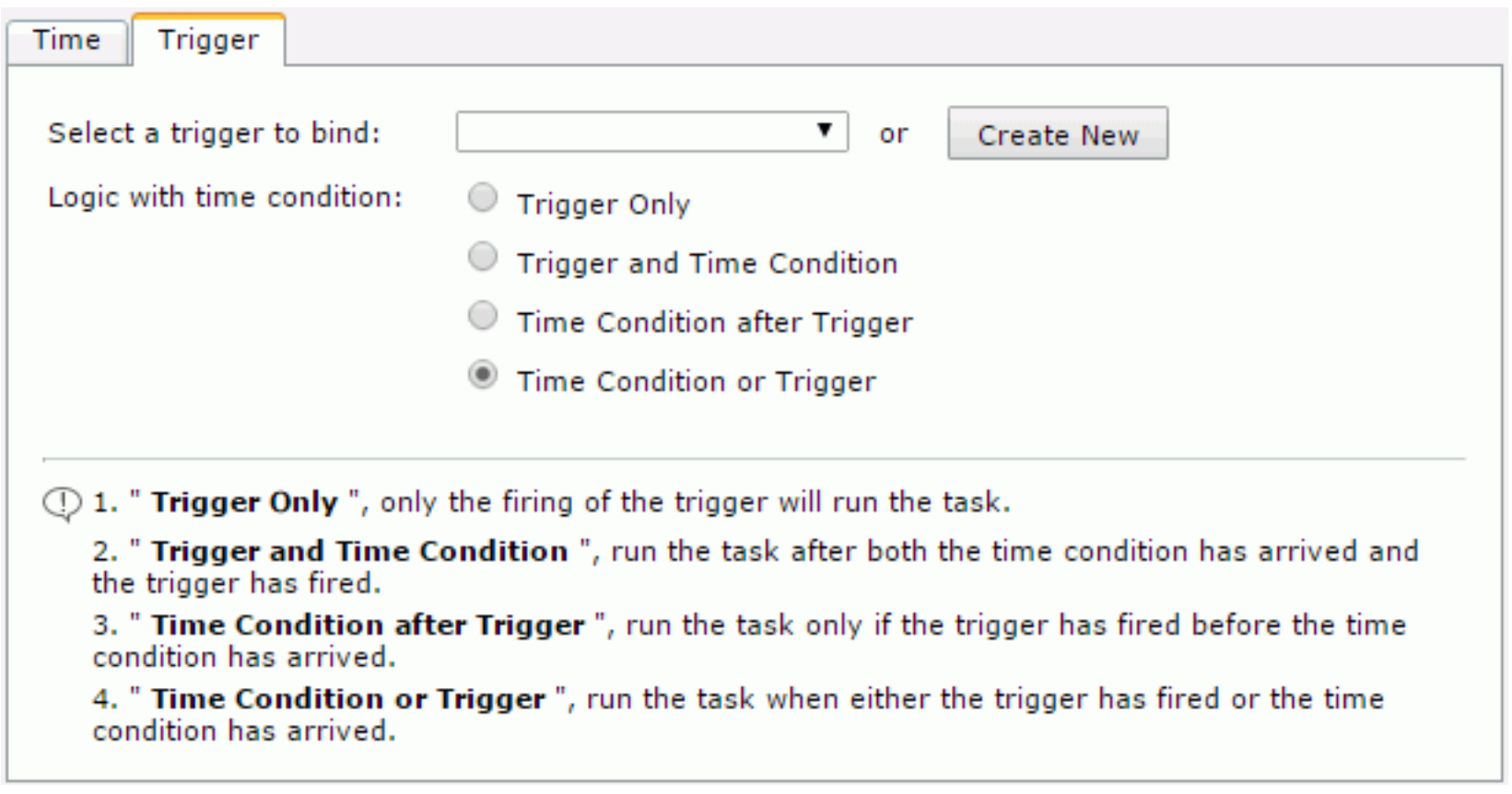
Time Type

Specifies the time type of the task.

- **Run this task immediately**
  Performs the task as soon as you submit it.
- **Run this task at**
  Specifies the time for when the task is to be performed.
  - **Date**
    Specifies the date for when the task will be run. You can specify the date by clicking the calendar button .
  - **Time**
    Specifies the time for when the task will be run.
  - **Run missed task upon server restart**
    Specifies to run missed tasks when you restart the server.
- **Run this task periodically**
  Specifies the time for when the task is to be performed on a repeated basis.
  - **Duration**
    - **Run after**
      Specifies the start date and time of the period during which the task is to be performed on a repeated basis.
    - **Run until**
      Specifies the end date and time of the period during which the task is to be performed on a repeated basis.
  - **Date**
    - **Daily**
      Performs the task every one day, two days, etc., or every weekday (from Monday to Friday).
    - **Weekly**
      Performs the task every one week, two weeks, etc., and on Sunday, Monday, Tuesday, etc.
    - **Monthly**
      Performs the task on the first day, second day, etc. of every one month, two months, etc., or on the first Sunday, Monday, etc. of every one month, two months, etc.
  - **Time**
    Specifies the exact time for when a task is to be performed on a selected day.
    - **At**
      Specifies a specific time for when to perform a task on a selected day.
    - **Hourly**
      Performs the task every one hour, two hours, etc., at a certain minute on a selected day.
    - **Minutely**
      Performs the task every one minute, two minutes, etc.
  - **Run missed task upon server restart**
    Specifies to run missed tasks when you restart the server.

Trigger tab

The Trigger tab allows you to set settings for specifying a trigger for the publishing task.



Select a trigger to bind

Specifies the trigger from the drop-down list for the task.

Create New

Creates a new trigger. This option is not available to organization users when the organization feature is enabled.

- **Trigger Name**
  Specifies the name of the trigger.
- **Conflict Check**
  Checks if the name has been used for another trigger.
- **Description**
  Specifies the description of the trigger (if required).
- **OK**
  Creates the trigger with the specified information.
- **Cancel**
  Cancels the trigger creating process.

Logic with time condition

Specifies the logic between time condition and trigger condition.

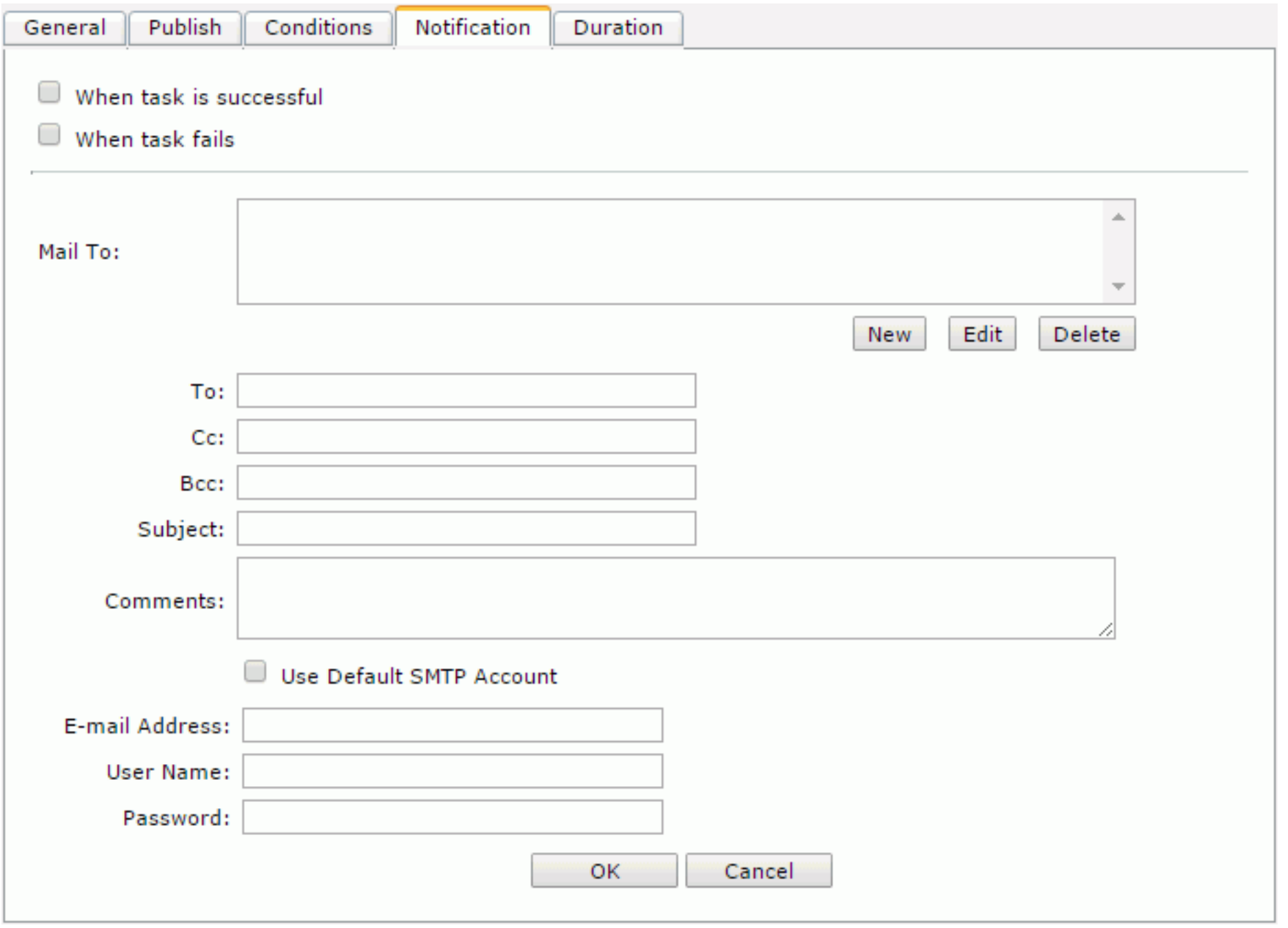
- **Trigger Only**
  Performs the task only when the trigger fires.
- **Trigger and Time Condition**
  Performs the task when both time condition and trigger condition are ready.

  Notes:

  - No matter which condition is ready, the task can only be performed when its counterpart is ready.
  - If you specify the task to be performed at a specific time, you must check the checkbox Run missed task upon Server restart, otherwise the task will be regarded as expired and will be deleted when the time condition is ready before the trigger condition.
- **Time Condition after Trigger**
  Performs a task when both the time condition and the trigger condition are ready. The trigger condition must be ready before the time condition. If the time condition is ready before the trigger condition, the task will be regarded as expired and will be deleted.
- **Time Condition or Trigger**
  Performs the task when either the time condition or the trigger condition is ready.

Notification tab

The Notification tab allows you to notify someone by e-mail when a task has finished running, regardless of whether it was successful or unsuccessful.



When task is successful

Specifies to send an e-mail when the task is successful.

When task fails

Specifies to send an e-mail when the task is unsuccessful.

Mail To

Lists the e-mail addresses you have sent mail to.

New

Creates a new e-mail.

- **To**
  Specifies the address you want to send the e-mail to.
- **Cc**
  Specifies the address you want to copy to.
- **Bcc**
  Specifies the address you want to secretly copy to.
- **Subject**
  Specifies the subject of the e-mail.
- **Comments**
  Specifies the contents of the e-mail or comments to the contents.
- **Use Default SMTP Account**
  Specifies whether to use the default SMTP account configured on the JReport Administration page > Configuration > E-mail Server panel as the e-mail publish account.

  The option is checked by default. Uncheck it and you will be able to specify a different account using the following options:

  - **E-mail Address**
    Specifies the e-mail address of the new SMTP account.
  - **User Name**
    Specifies the user name of the new account.
  - **Password**
    Specifies the password of the new account.
- **OK**
  Retains the settings and adds the e-mail address into the Mail To list.
- **Cancel**
  Cancels operations.

Edit

Edits the specified e-mail in the Mail To list.

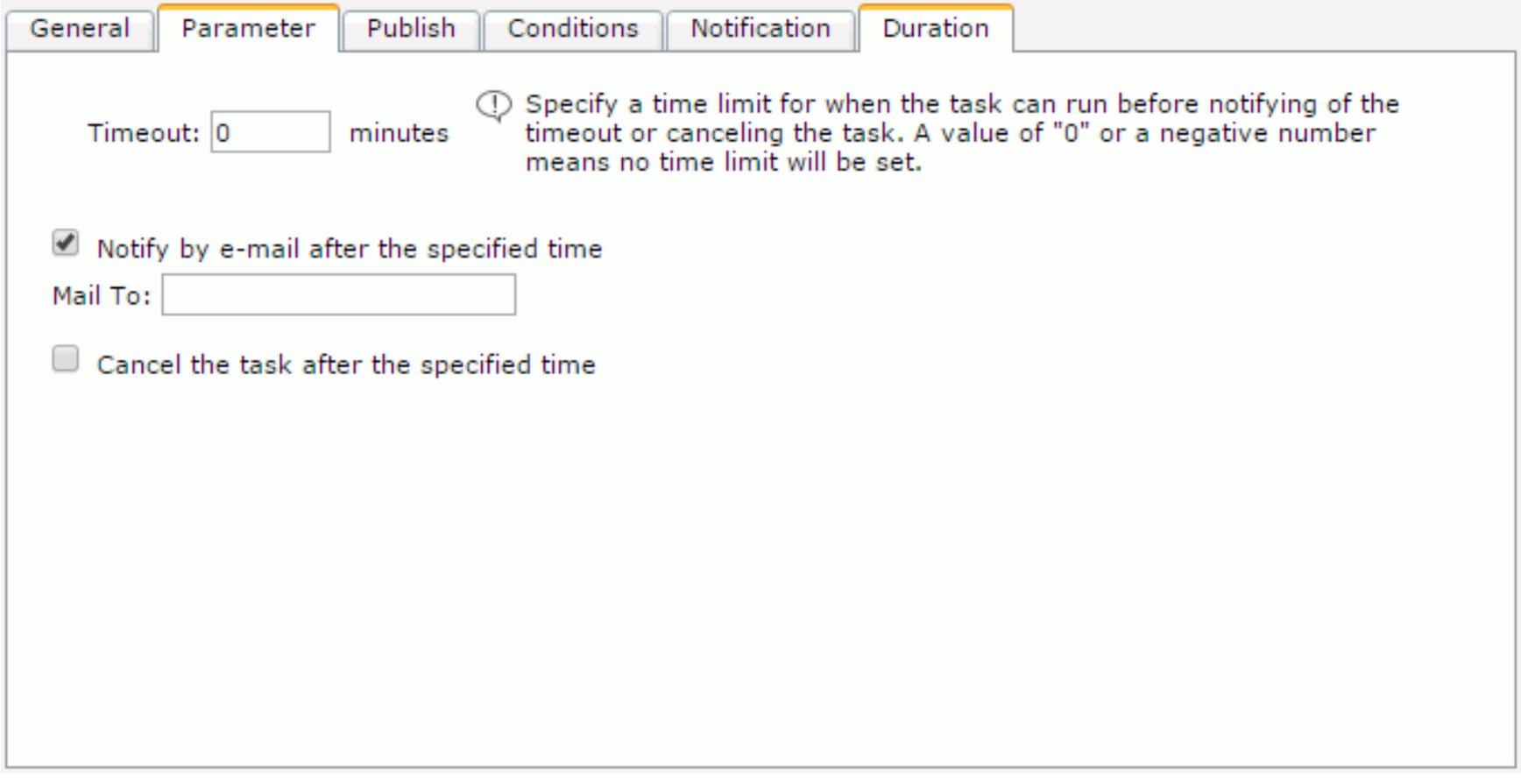
Delete

Deletes the specified e-mail from the Mail To list.

Duration tab

This tab is available only when you have enabled the task-level timeout mechanism either by setting the task.duration.enable property to true in the server.properties file or by checking the Enable Task Duration option in the Configuration > Advanced panel on the JReport Administration page.

In this tab, you can specify a time duration for a task, and ask JReport Server to cancel the task or to notify you or someone else of the task status via e-mail if the task has not yet finished running when the task duration is up.



The following are the available options:

Timeout

Specifies the time duration for the task.

Notify by e-mail after the specified time

Sends an e-mail about the task information when the specified time is up.

- **Mail To**
  Specifies the e-mail address of the recipient.

Cancel the task after the specified time

Cancels the running task when the specified time is up.

Security dialog

When you click Security on the system toolbar of the JReport Administration page, the entries to the following panels are available on the drop-down menu. Use each panel to set the settings according to your requirements.

- Realm panel
- User panel
- Group panel
- Role panel
- Privilege panel
- Alias panel
- Dynamic Security panel

Realm panel

This panel allows you to manage realms in JReport Server.

Create a New Realm
Creates a realm in the New Realm dialog.

Name
Displays the names of the existing realms.

State
Displays the states of the existing realms: Active Realm or Selected Realm. Active Realm means that the realm is the current active realm, and Selected Realm means the realm is the current editing realm in the Security dialog.

Control
Controls the realms.

- **Select**
  Sets the specified realm to be the current editing realm.
- **Delete**
  Deletes the specified realm.

User panel

This panel allows you to manage users in JReport Server.

Create a New User
Opens the New User dialog to create a user.

User Name
Lists the users' names. You can view and edit user properties by clicking the underlined user names.

Full Name
Displays the user's full names. Full name is a property of a user.

Edit Roles
Edits the roles of the specified user. Click the underlined role(s) to edit the roles.

- **Name**
  Lists the name of the roles that the user holds.
- **Built-in**
  Shows whether the role is a built-in role.
- **Remove**
  Removes the specified roles from the role list.
- **Add Roles**
  Lists the roles that can be assigned to the user.
  - **Add**
    Adds the specified roles to the user.

Edit Groups
Edits the groups of the specific user. Click the underlined group(s) to edit the groups.

- **Name**
  Lists the name of the groups that the user belongs to.
- **Remove**
  Removes the user from the specified groups.
- **Add Groups**
  Lists the groups that the user can be added to.
  - **Add**
    Adds the user to the specified groups.

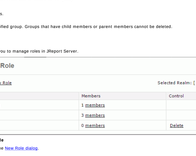
Authentication
Specifies users' authentication type: Local or LDAP.

Control
Controls the users.

- **Auditing**
  Opens the Auditing dialog to audit the specified user.
- **Change Password**
  Opens the Change Password dialog to change the password of the specific user.
- **Preferences**
  Opens the Preferences dialog to set preferences for the specific user. For details about the preference settings, refer to Profile dialog.
- **Delete**
  Deletes the specified user. The built-in user accounts admin and guest cannot be deleted.

Group panel

This panel allows you to manage groups in JReport Server.



Create a New Group
Creates a group in the New Group dialog.

Name
Lists the group names. You can view and edit group properties by clicking the underlined group name.

Members
Edits members of the specified group. Click the underlined member(s) to edit the members.

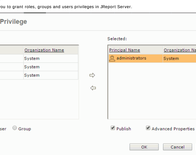
- **Name**
  Lists the name of the members that the group holds.
- **Type**
  Specifies the type of the members that the group holds.
- **Authentication**
  Specifies the group members' authentication type: Local or LDAP.
- **Remove**
  Removes the specified members from the group.
- **Add Members**
  Shows the members that can be added to the group.

Control
Controls the groups.

- **Delete**
  Deletes the specified group. Groups that have child members or parent members cannot be deleted.

Role panel

This panel allows you to manage roles in JReport Server.



Create a New Role
Creates a role in the New Role dialog.

Name
Lists the role names. You can view and edit role properties by clicking the underlined role name.

Members
Edits members of the specified role. Click the underlined member(s) to edit the members.

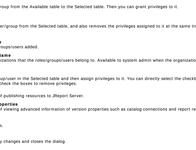
- **Name**
  Lists the names of the role's members.
- **Type**
  Specifies the type of the role's members.
- **Authentication**
  Specifies the role's members' authentication type: Local or LDAP.
- **Remove**
  Removes the specified members from the role.
- **Add Members**
  Shows the members that can be added to the role.

Control
Controls the roles.

- **Delete**
  Deletes the specified role. Built-in roles, administrators and everyone, and roles that are not empty, cannot be deleted.

Privilege panel

This panel allows you to grant roles, groups and users privileges in JReport Server.



Available

- **Role**
  Lists the roles available in the server security system.
  - **Role Name**
    Displays the names of the roles.
  - **Organization Name**
    Shows the organizations that the roles belong to. Available to system admin when the organization feature is enabled.
- **User**
  Lists the users available in the server security system.
  - **User Name**
    Displays the names of the users.
  - **Organization Name**
    Shows the organizations that the users belong to. Available to system admin when the organization feature is enabled.
- **Group**
  Lists the groups available in the server security system.
  - **Group Name**
    Displays the names of the groups.
  - **Organization Name**
    Shows the organizations that the groups belong to. Available to system admin when the organization feature is enabled.

Adds a role/user/group from the Available table to the Selected table. Then you can grant privileges to it.

Removes a role/user/group from the Selected table, and also removes the privileges assigned to it at the same time.

Selected

- **Principal Name**
  Lists the roles/groups/users added.
- **Organization Name**
  Shows the organizations that the roles/groups/users belong to. Available to system admin when the organization feature is enabled.
- **Privileges**
  Shows the privileges available in the server security system.
  Select a role/group/user in the Selected table and then assign privileges to it. You can directly select the checkboxes to grant privileges, or uncheck the boxes to remove privileges.
  - **Publish**
    The privilege of publishing resources to JReport Server.
  - **Advanced Properties**
    The privilege of viewing advanced information of version properties such as catalog connections and report related resources.

OK
Applies the settings.

Cancel
Does not retain any changes and closes the dialog.

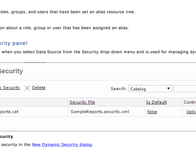
Help
Displays the help document of this feature.

Notes:

- If users want to publish resources, they must at the same time have both the permission to write and publish.
- By default, only the "administrators" role and "admin" users have the Publish and Advanced Properties privileges.

Alias panel

This panel allows you to set an alias resource tree based on the server resource tree for roles, groups, or users.



Set Role/Group/User Alias
Sets an alias resource tree for roles/groups/users, and then adds these roles/groups/users to the alias list.

After clicking the link, the following options will be available:

- **Role Name/Group Name/User Name**
  Lists the roles, groups, and users for which you can set an alias resource tree.
- **Description**
  Specifies information about a role, group or user.
- **Next**
  Goes to the next step and opens the Set Alias dialog.
- **Cancel**
  Cancels any operations performed and exits the page.

Name
Lists the name of roles, groups, and users that have been set an alias resource tree.

Description
Specifies information about a role, group or user that has been assigned an alias.

Dynamic Security panel

This panel appears when you select Data Source from the Security drop-down menu and is used for managing dynamic catalog security.

New Dynamic Security
Creates a dynamic security in the New Dynamic Security dialog.

Delete
Deletes the selected dynamic security.

Search
Searches for specific dynamic security.

- **Search category drop-down list**
  Specifies the search category. For example, if you want to find the dynamic security which use the catalog SampleReports.cat, select Catalog from the drop-down list.
  There are the following search categories:
  - **Catalog**
    Searches for catalogs.
  - **Security File**
    Searches for catalog security file names.
  - **Is Default**
    Searches for true or false in the Is Default column.
- **Keyword text box**
  Specifies the search keyword. After inputting the text, click in the text box to start the search. The button in the text box is used to clear the input text.

Dynamic security table
Lists all the dynamic security. All columns are sortable. To sort the table by a certain column, click on the column name. By Default the table is sorted by Catalog.

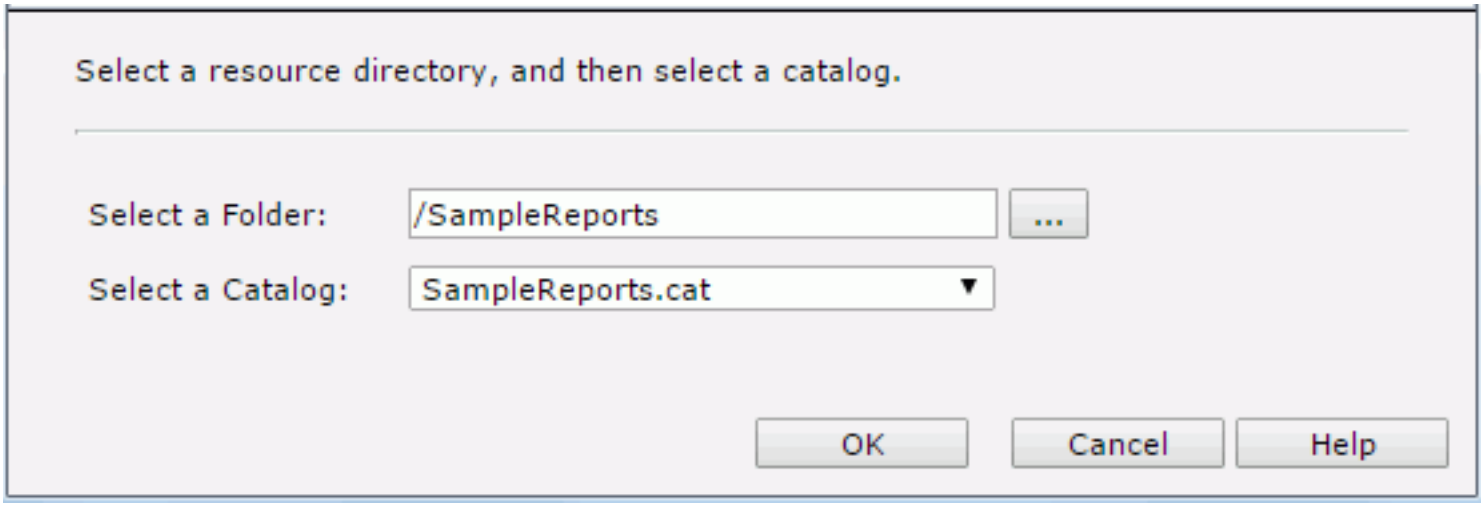
- **Checkbox**
  Specifies whether or not to select a dynamic security. Select the checkbox on the column header to select all the dynamic security.
- **Catalog**
  Displays the catalogs for which the dynamic security are created with the full resource path, for example, /SampleReports/GeneralReports.cat.
- **Security File**
  Displays the names of the catalog security files for the dynamic security.
- **Is Default**
  Shows whether the dynamic security is set to the default. The value is editable. Click in the Is Default cell of a dynamic security and then select true or false from the drop-down list. For a catalog, only one dynamic security can be set to true and the latest setting takes effect.
- **Controls**
  - **Upload**
    Uploads a catalog security file for a specific dynamic security in the Upload Dynamic Security dialog.
  - **Download**
    Download the catalog security file from the server.

Related topics:

- Managing security

Select Another Catalog dialog

This dialog helps you to specify another catalog for the current report/library component or folder.



Select a Folder

Specifies the folder in the server resource tree that contains the required catalog. You can click the button ... to open the Select Folder dialog for selecting the folder.

Select a Catalog

Lists the catalogs in the specified folder. Select the required one from the drop-down list.

OK

Accepts the settings and closes the dialog.

Cancel

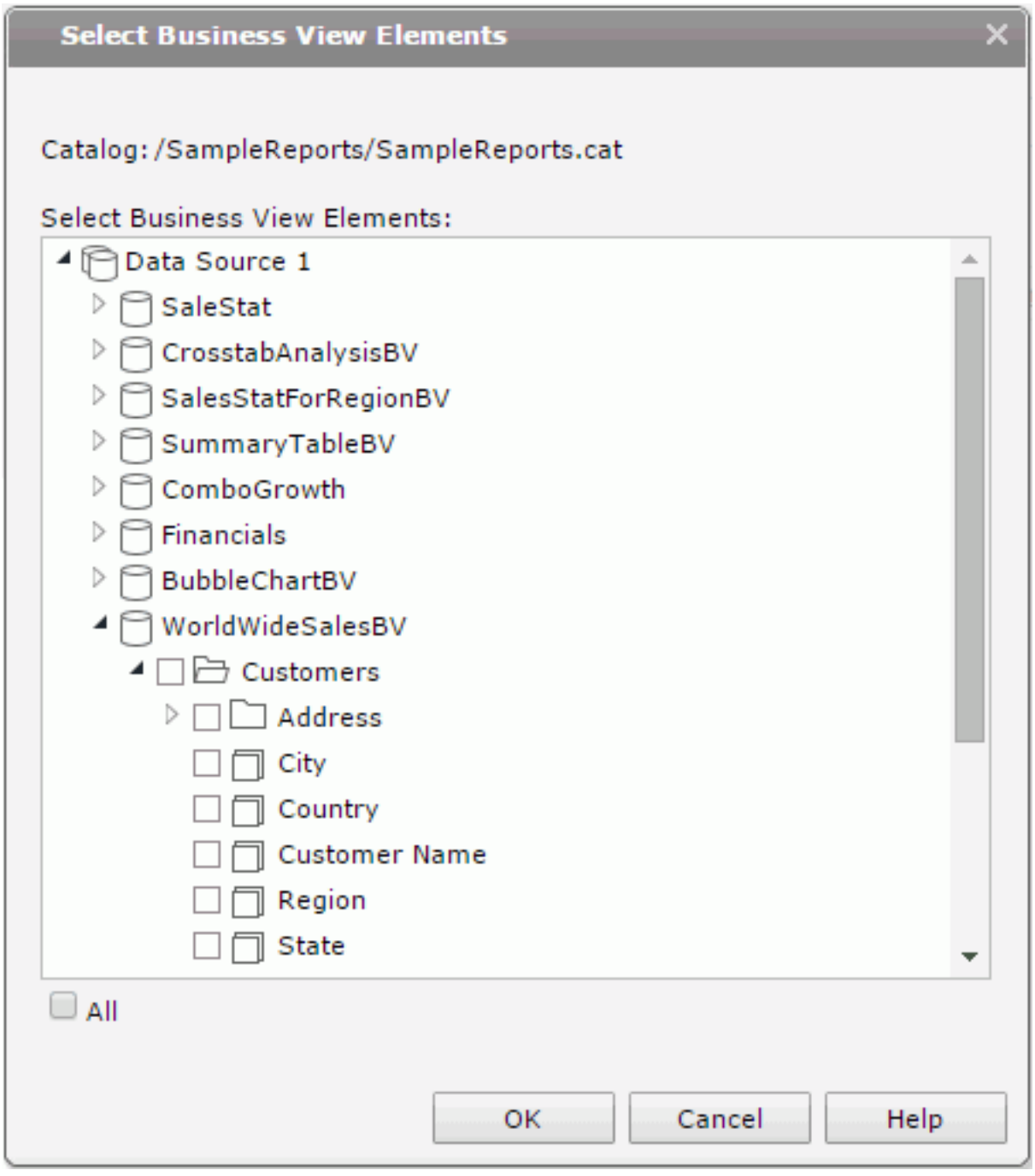
Cancels the settings and closes the dialog.

Help

Displays the help document about this feature.

Select Business View Elements dialog

The dialog appears when you click the Select Business View Elements link in the Add Dynamic Display Names dialog. It helps you to select business view elements in the specified catalog for which to define the dynamic display names.



Catalog

Displays the current catalog with the resource path.

Select Business View Elements

The panel lists all the business views and their elements in the catalog. You can select multiple elements by selecting the corresponding checkboxes.

All

Selects/Deselects all the business view elements in the catalog.

OK

Applies the selection and exits the dialog.

Cancel

Cancels the selection and closes the dialog.

Help

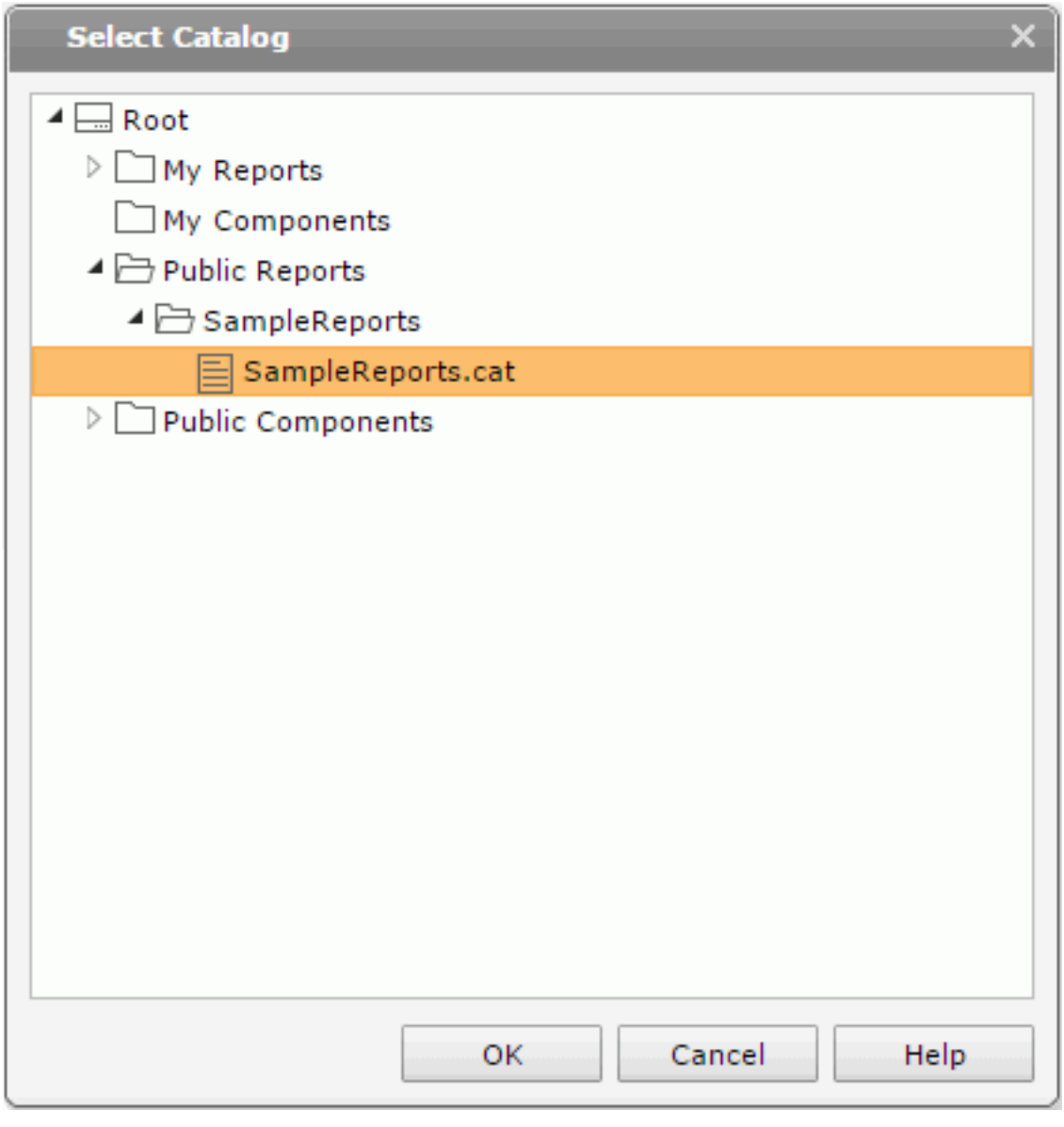
Displays the help document about this feature.

Related topics:

- Managing dynamic display names of business view elements

Select Catalog dialog

This dialog helps you to select a catalog in the server resource tree.



Resource tree box

The box lists the folder and catalog resources in the server resource tree. Navigate to select the required catalog.

OK

Selects the catalog and exits the dialog.

Cancel

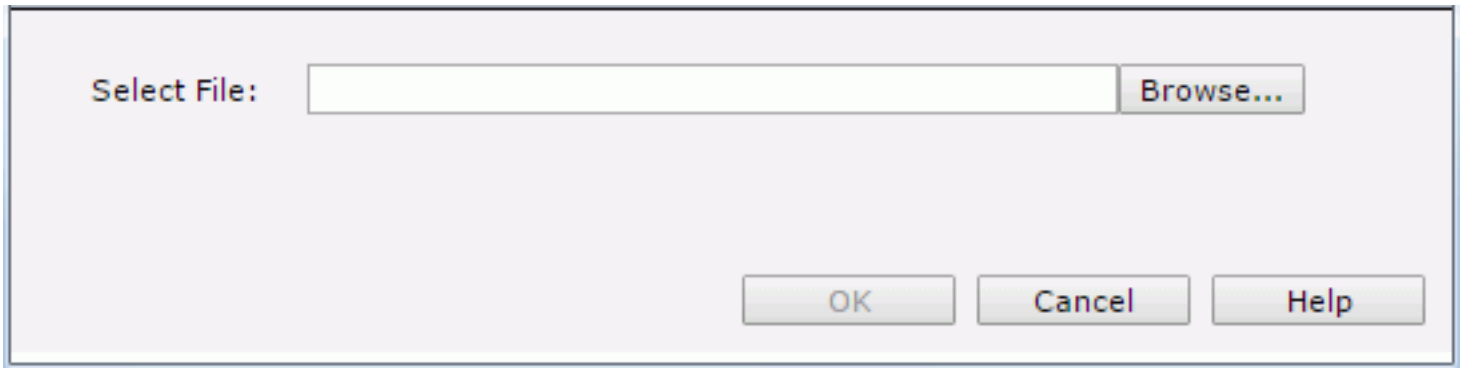
Cancels the selection of a catalog and closes the dialog.

Help

Displays the help document about this feature.

Select Dynamic Display Name File dialog

This dialog appears when you click the Import link in the Dynamic Display Names column in the Dynamic Display Names panel. It helps you to import dynamic display names from a local resource bundle file.



Select File

Specifies the file from which to import the dynamic display names.

Click the **Browse** button to select the local resource bundle file that defines the dynamic display names of the business view elements in a catalog.

OK

Imports the dynamic display names from the selected file and exits the dialog.

Cancel

Cancels the importation of dynamic display names and closes the dialog.

Help

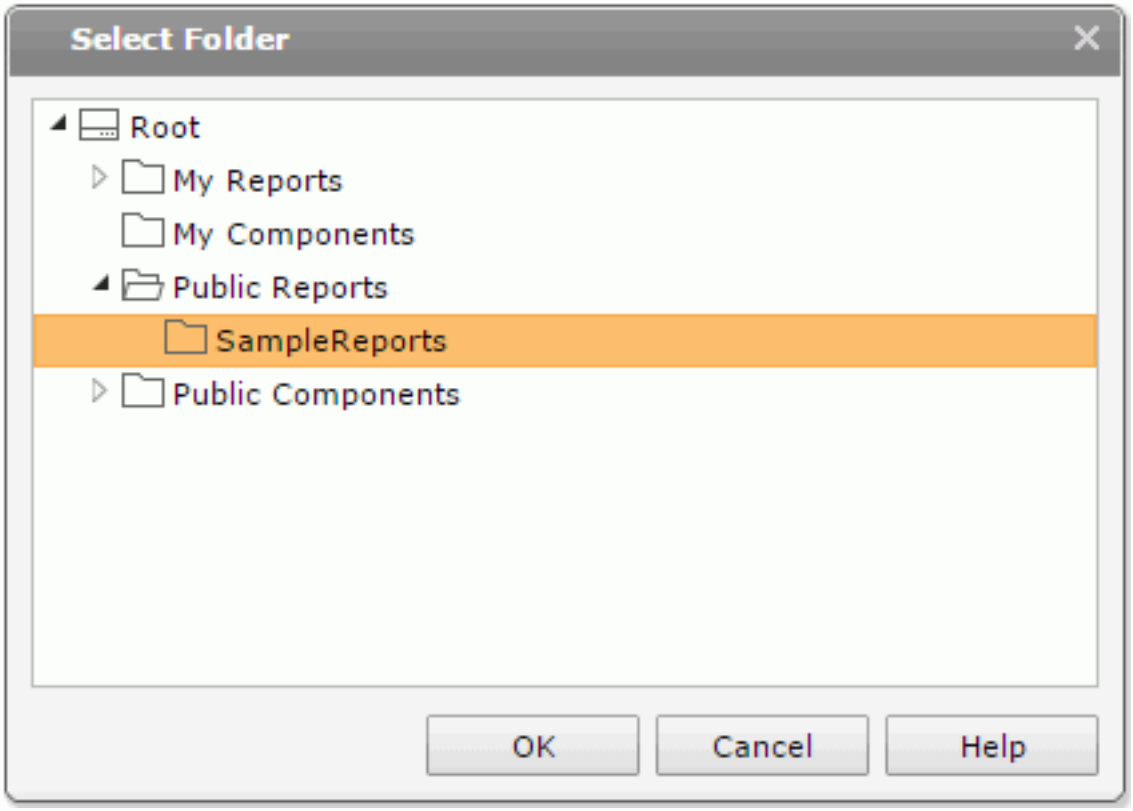
Displays the help document about this feature.

Related topics:

- Managing dynamic display names of business view elements

Select Folder dialog

This dialog helps you to select a folder in the server resource tree that contains the required resource.



Resource tree box

The box lists the folder resources in the server resource tree. Navigate to the required folder to select it.

OK

Selects the folder and closes the dialog.

Cancel

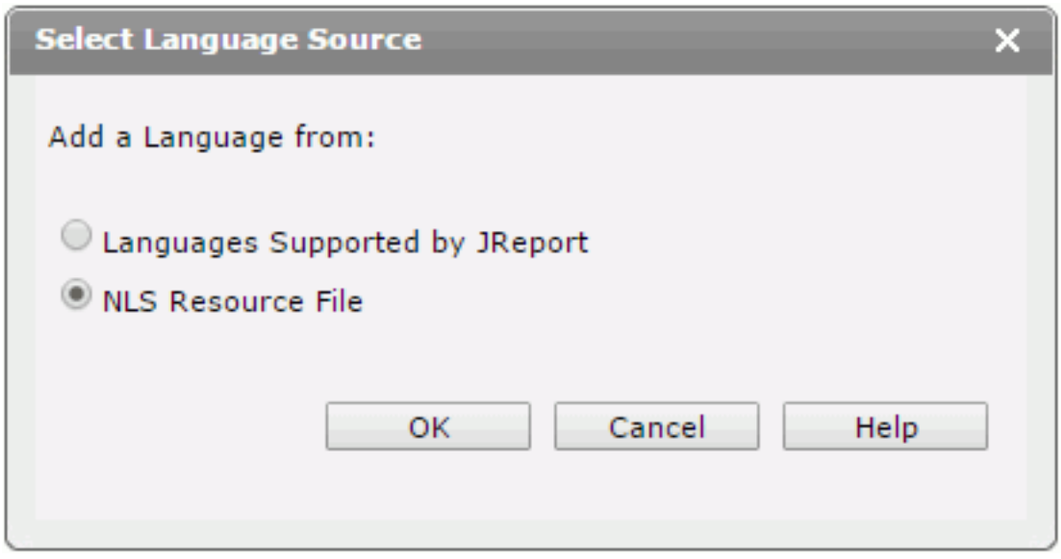
Cancels the selection of a folder and closes the dialog.

Help

Displays the help document about this feature.

Select Language Source dialog

This dialog appears when you click the button + above the Language box in the Global NLS dialog, Resource Tree NLS dialog or the NLS Editor dialog when editing NLS for a catalog. It helps you to choose where to add new languages.



Languages Supported by JReport

Specifies to add new languages from the Available Languages box. After choosing the option and clicking OK, you will be shown the Add Language dialog.

NLS Resource File

Specifies to add new languages from NLS resource files in your local folder. After choosing the option and clicking OK, you can browse to the local folder where the NLS resource files are located.

NLS resource files should follow the naming rule: NLS_[language]_[region A2]_[User Defined]. properties. The language argument is a valid ISO Language Code as defined by ISO-639. You can find a full list of these codes at a number of sites, for example: http://www.loc.gov/standards/iso639-2/php/code_list.php. The region argument is a valid ISO Country Code as defined by ISO-3166. You can find a full list of these codes at a number of sites, for example: http://www.chemie.fu-berlin.de/diverse/doc/ISO_3166.html.

OK

Goes to the next action.

Cancel

Cancels the action and exits the dialog.

Help

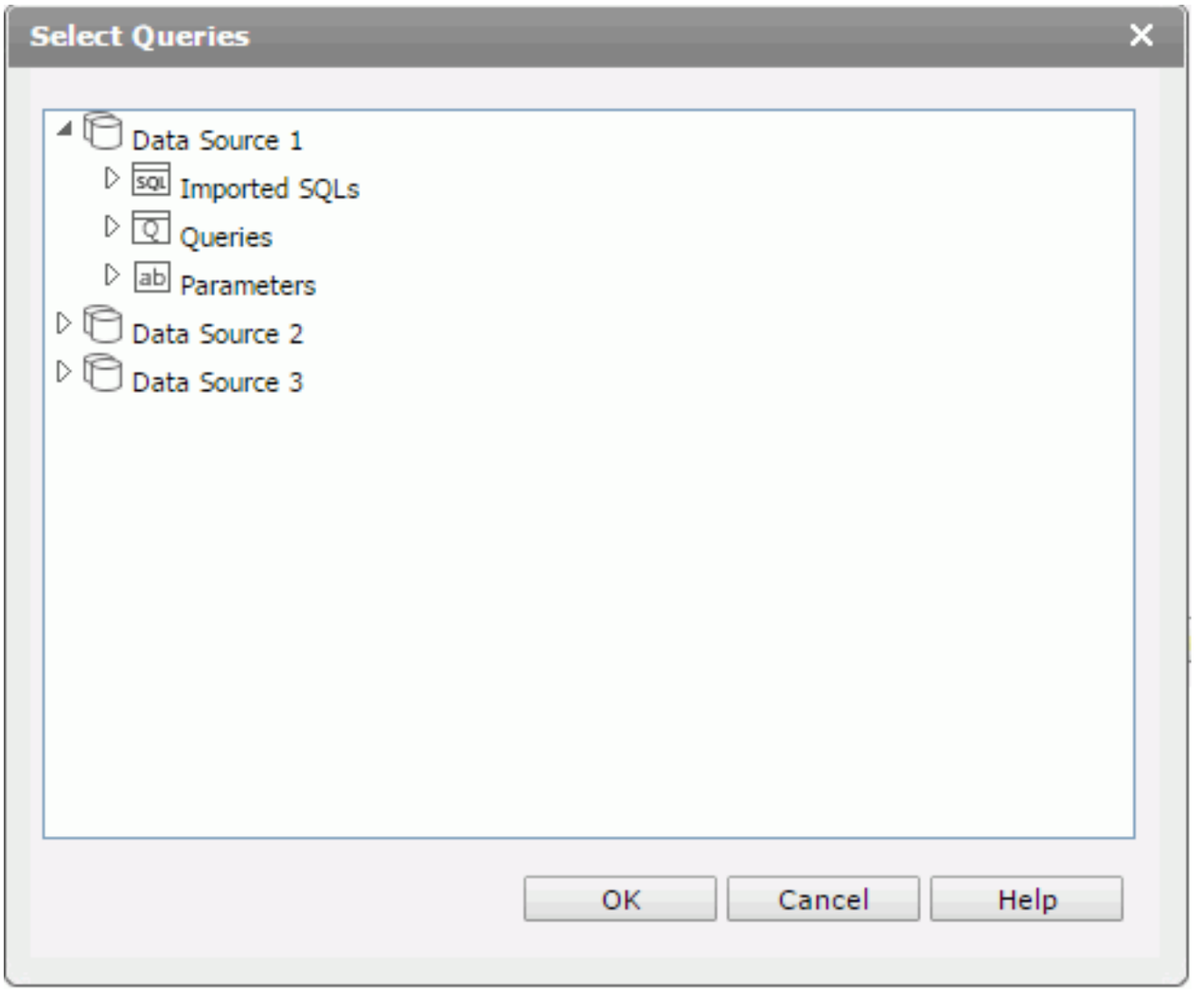
Displays the help document about this feature.

Related topics:

- Creating global NLS
- Editing local NLS
- Editing resource tree NLS

Select Queries dialog

This dialog appears when you click the button + beside the Select Queries box in the New Cache dialog for selecting queries. It allows you to specify the queries that you are going to create data caches for.



Resource tree box

The box lists the data source resources that can be used as queries. The query resources include queries, stored procedures, imported SQLs, user defined data sources, hierarchical data sources, and parameters whose type is Bind with Single Column or Bind with Cascading Columns (for which type only a parameter with the cascading parameter group name is listed). If a query resource has been used by an existing data cache, it will not be listed here. You can select multiple resources at a time.

OK

Selects the query resources and closes the dialog.

Cancel

Cancels the selection of resources and closes the dialog.

Help

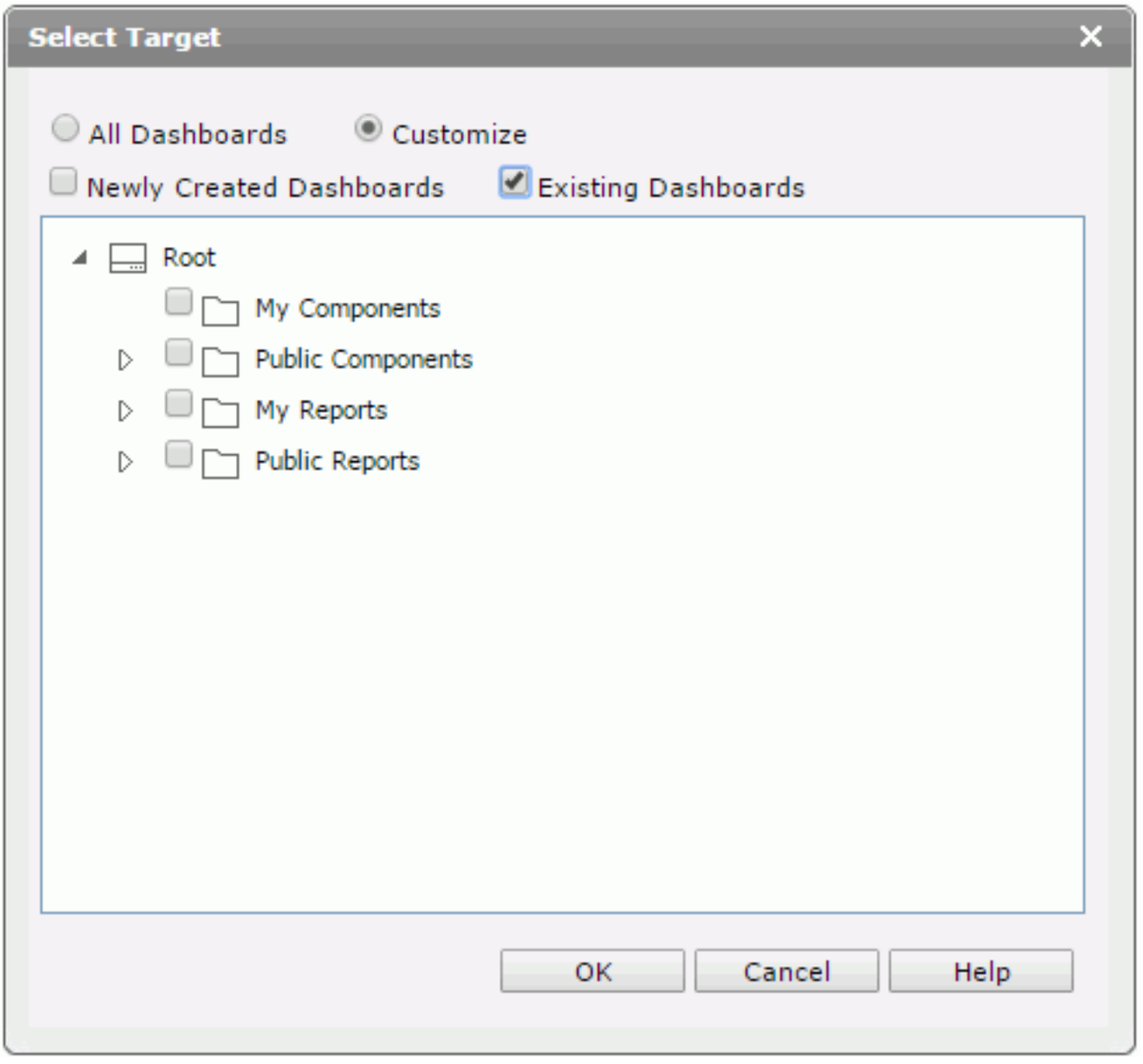
Displays the help document about this feature.

Related topics:

- Managing cached report data

Select Target dialog

This dialog is used for specifying the target resources you would like the implementation to apply to.



All Dashboards

If selected, the implementation will apply to all dashboards including the newly created ones that have not been saved yet.

Customize

If selected, you can specify the dashboards you want the implementation to apply to.

- **Newly Created Dashboards**
  If selected, the implementation will apply to all newly created dashboards that have not been saved yet.
- **Existing Dashboards**
  If selected, you can further specify one or more folders in which the implementation will apply to all of the dashboards.

OK

Applies the setting and exits the dialog.

Cancel

Cancels the setting and closes the dialog.

Help

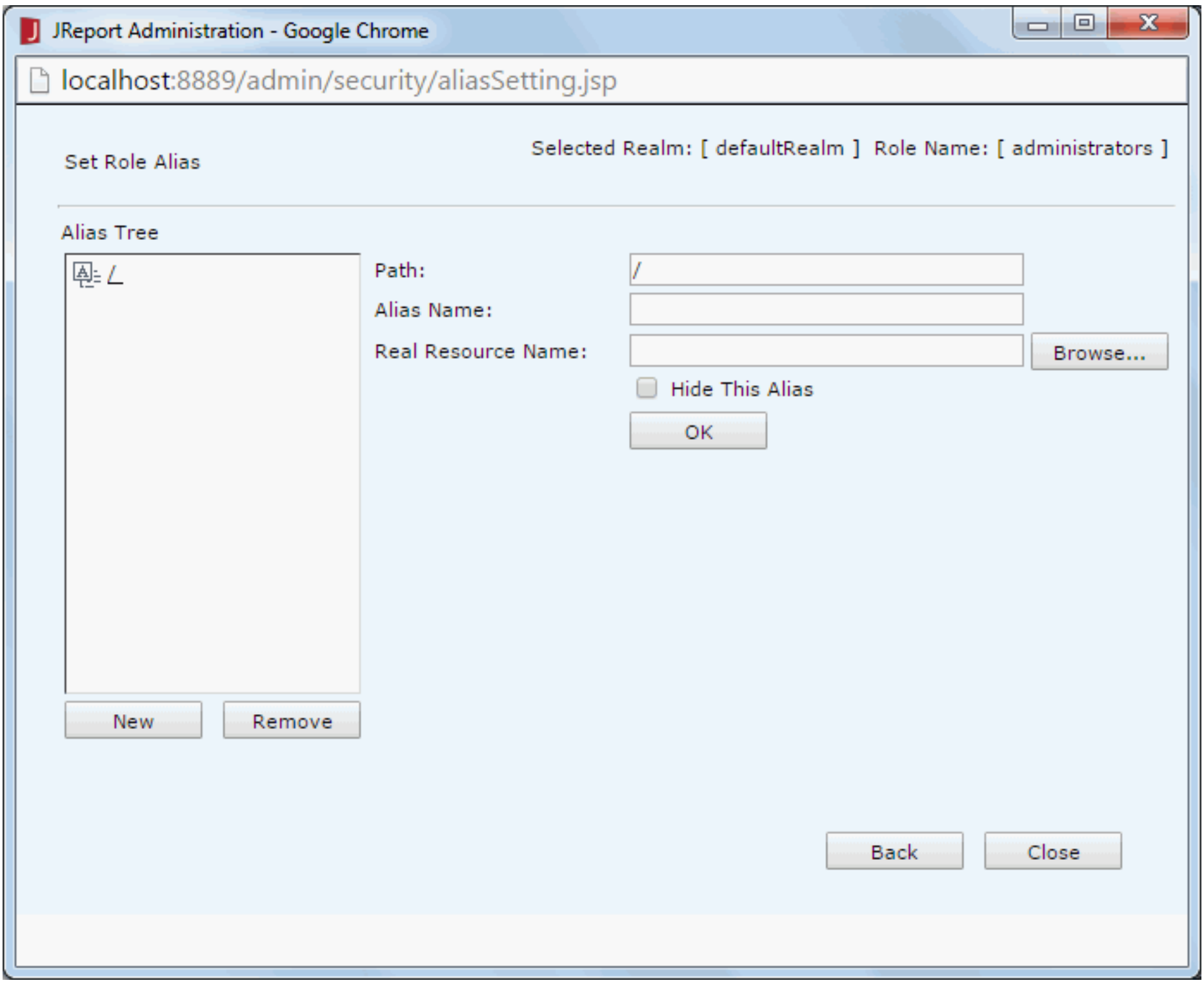
Displays the help document about this feature.

Related topics:

- Dashboard Listener API

Set Alias dialog

This dialog appears when you click the Set Alias link in the Security > Alias panel on the JReport Administration page, check a role/group/user in the displayed dialog and then click the Next button.



Alias Tree

Displays the alias resource tree.

New

Creates a new alias node in the current node. A node functions as a folder.

Remove

Deletes the selected alias node permanently. The root node cannot be deleted.

Path

Displays the path of the current node in the alias resource tree.

Alias Name

Displays the name of the current node. You can edit the alias name.

When you click the New button to create a new alias node in the current node, the default name for the new alias node is newAlias.

Real Resource Name

Displays the resource in the server resource tree that is associated with the current alias node. You can change the resource using the Browse button.

When you click the New button to create a new alias node in the current node, this field is left blank. You will need to specify a resource from the server resource tree with which to associate the new alias node by using the Browse button.

Browse

Specifies a destination resource in the server resource tree that is to be associated with the current alias node.

Hide This Alias

If this option is selected, the current alias node and its sub aliases will be invisible to the role, group, or user for whom the alias resource tree is set.

OK

Applies the settings.

Back

Returns to the previous page if no alias has been set.

Close

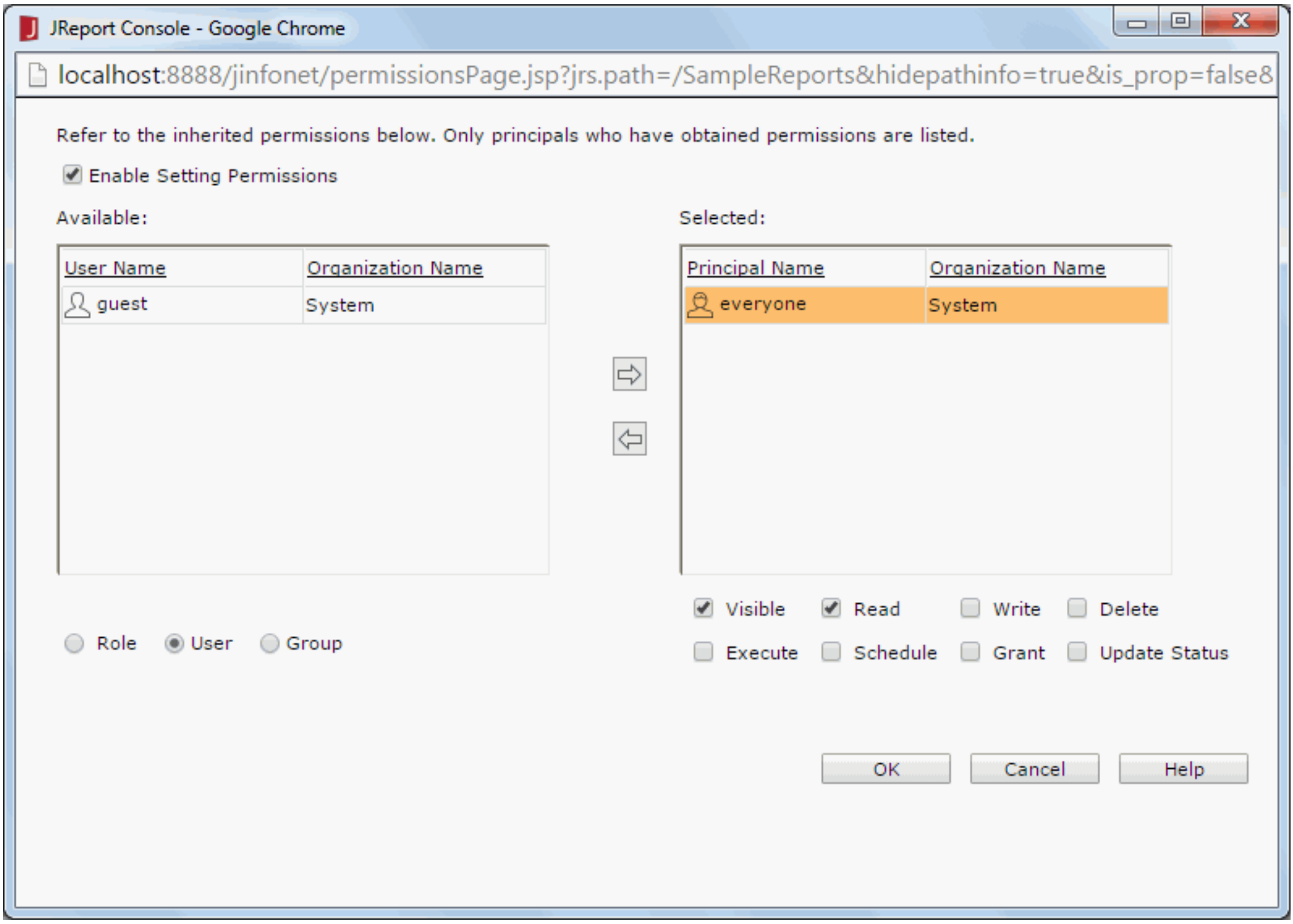
Exits this dialog.

Notes:

- An alias tree is based on the resource nodes (not virtual resource nodes) of the resource tree.
- By default, the alias resource tree root for each user refers to the resource tree root.
- When an alias tree is activated for a user, all resource access is then controlled by the alias resource tree.

Set Permissions dialog

This dialog helps you to assign permissions to roles, users, groups on specific resource or folder.



Enable Setting Permissions

Enables the setting of permissions.

Available

Lists the roles/users/groups to which you can assign permissions. When organization is enabled, the Organization Name column will be available in the table.

- **Role**
  If checked, all roles will be displayed in the Available box for you to assign permissions.
- **User**
  If checked, all users will be displayed in the Available box for you to assign permissions.
- **Group**
  If checked, all groups will be displayed in the Available box for you to assign permissions.

Adds the selected role, user or group to the Selected box.

Removes the selected role, user or group from the Selected box.

Selected

Select a role/user/group in the Selected box and then select the permissions you would like the role/user/group to have on the specific resource or folder. For what each permission is used for, see Permission.

When organization is enabled, the Organization Name column will be available in the table.

OK

Accepts the settings and closes the dialog.

Cancel

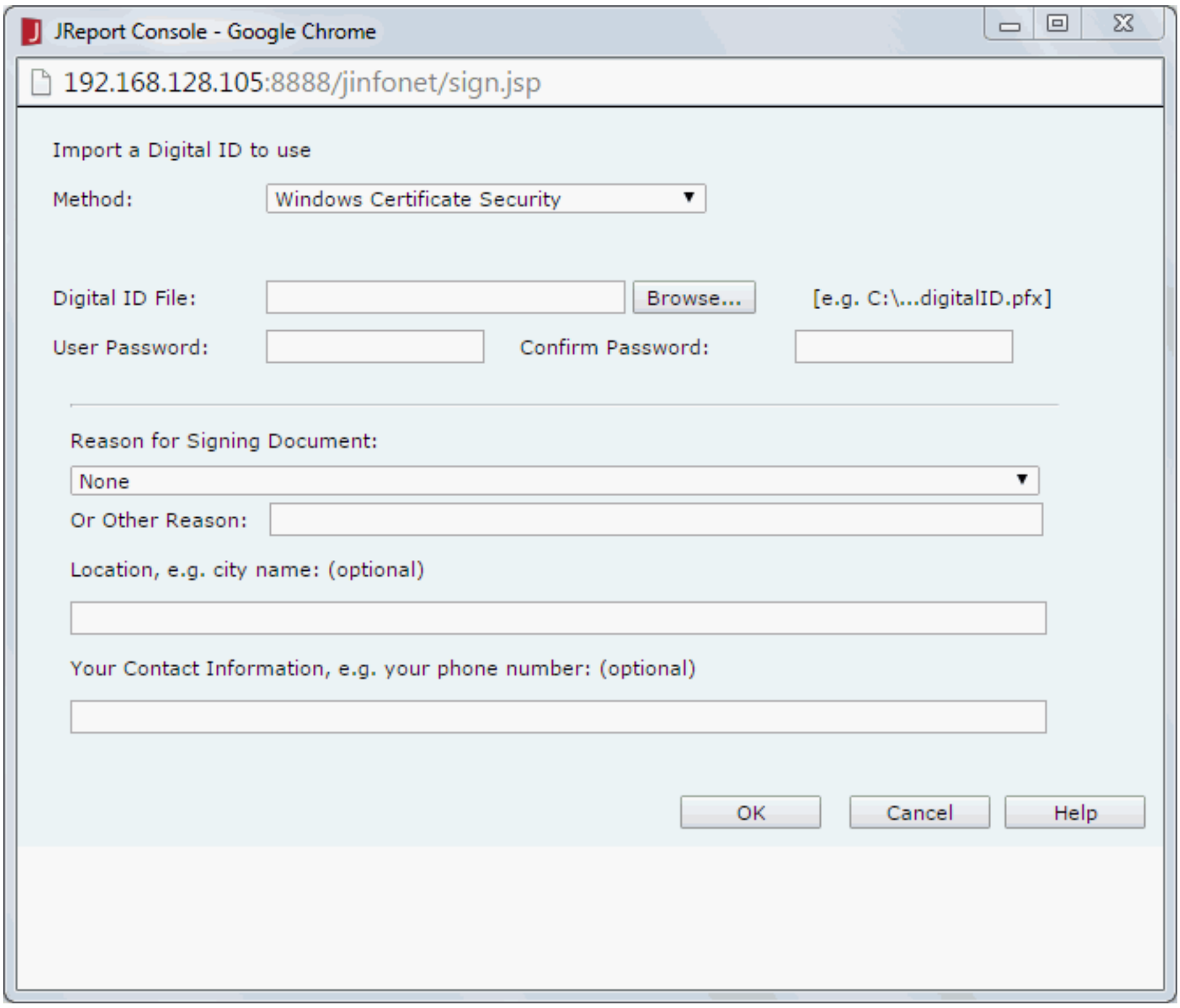
Cancel the settings.

Help

Displays the help document about this feature.

Sign dialog

This dialog appears when you click the Setting button next to the Sign option while configuring settings for advanced running/publishing a report in PDF format.



Method

Specifies the signing digital signature method from the three methods provided by Adobe Acrobat software.

Digital ID File

Specifies whether to use a digital ID file from a specified directory.

User Password

Specifies the password for this digital ID file.

Confirm Password

Confirms the password.

Reason for Signing Document

Specifies the reason for signing the document.

Or Other Reason

Edits the reason for signing the document.

Location

Edits your location accordingly.

Your Contact Information

Edits your contact information accordingly.

OK

Saves the settings and exits the dialog.

Cancel

Cancels the settings and closes the dialog.

Help

Displays the help document about this feature.

Triggers dialog

This dialog appears when you click Triggers on the system toolbar of the JReport Administration page.

New Trigger

Creates a new trigger.

- **Trigger Name**
  Specifies the unique name for the new trigger. Triggers are managed by name in JReport Server, so each trigger must have a unique name.
- **Conflict Check**
  Checks whether the name conflicts with any existing triggers before creating the new trigger. You cannot create two triggers with the same name.
- **Description**
  Specifies the description for the new trigger.
- **OK**
  Creates the trigger.
- **Cancel**
  Cancels the operation.

Delete

Deletes the selected triggers.

Enable

Enables the selected triggers. When a trigger is enabled, the Fire operation is also enabled.

Disable

Disables the selected triggers.

Fire

Activates tasks that are bound with the selected triggers.

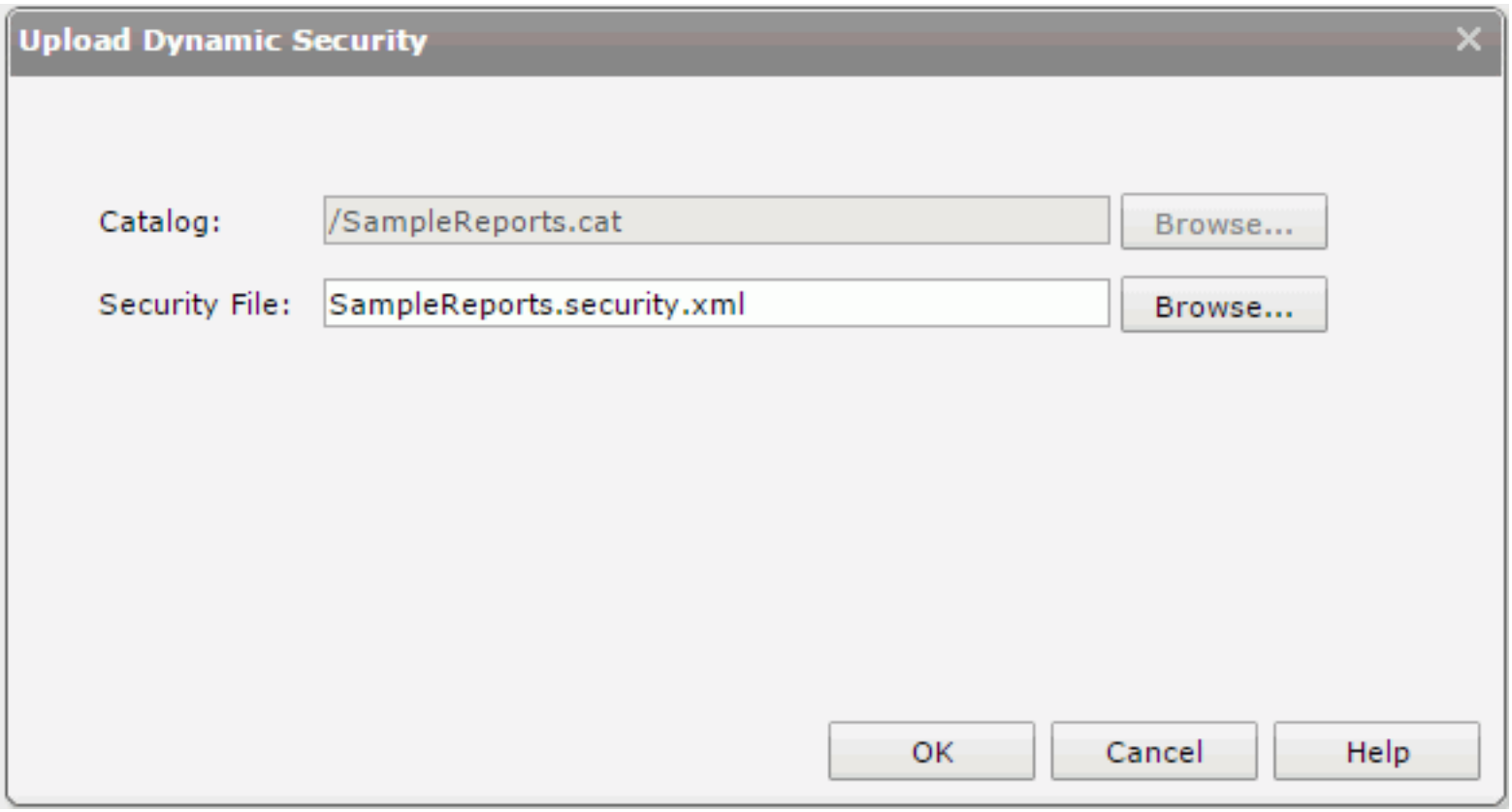
Trigger box

Displays the trigger information.

- **Checkbox**
  Specifies whether or not to select the triggers. Check the checkbox on the column header to select all the triggers.
- **Name**
  Displays names of the triggers.
- **Is Enabled**
  Shows whether the triggers are enabled or not.
- **Referenced**
  Shows the times the triggers are referenced.
- **Last Fired**
  Shows the time when the triggers were lasted fired.
- **Description**
  Displays description of the triggers.

Upload Dynamic Security dialog

The dialog appears when you click the Upload link in the Dynamic Security panel. It helps you to upload a catalog security file for a specific dynamic security.



Catalog

Displays the catalog for which the dynamic security is created.

Security File

Specifies another security file to replace the current one. Click the Browse button to select a catalog security file that defines member-level security, record-level security and column-level security in the catalog.

OK

Uses the new security file to create dynamic security for the catalog.

Cancel

Cancels the uploading of a security file and closes the dialog.

Help

Displays the help document of this feature.

Related topics:

- Managing dynamic security

Visual Analysis Template Properties dialog

You can set the properties of a Visual Analysis template in its Properties dialog according to your requirements.

The Properties dialog is divided into the following two tabs:

- General
- Permission

Reset

Discards your modifications and restores the dialog to its default status.

OK

Retains the settings and submits the task to server.

Cancel

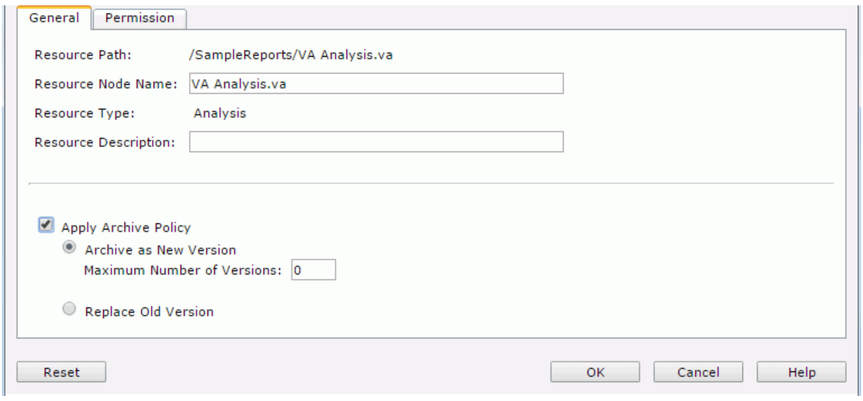
Cancels any settings and closes the dialog.

Help

Displays the help document about this feature.

General tab

Specifies the general properties of the Visual Analysis template.



Resource Path

Shows the resource path.

Resource Node Name

Specifies the name for the Visual Analysis template.

Resource Type

Shows the type of the resource.

Resource Description

Specifies the description for the Visual Analysis template.

National Language Support

Specifies whether to enable the NLS feature for the Visual Analysis template. This option is available when the Properties dialog is opened from the JReport Administration page.

[Custom Field Name]

Specifies value of the custom field for the Visual Analysis template. A custom field can be regarded as a resource property and is available when it is enabled.

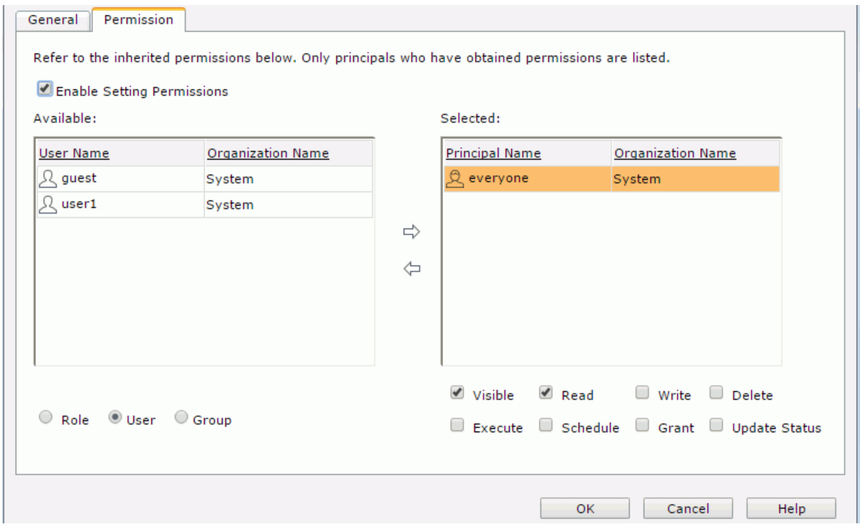
Apply Archive Policy

Applies an archive policy to the Visual Analysis template versions.

- **Archive as New Version**
  Specifies whether to use multiple versions for the Visual Analysis template. Available only for versions in the resource tree.
  - **Maximum Number of Versions**
    Specifies the maximum version amount that will be listed in the version table. By default the version amount is unlimited.
- **Replace Old Version**
  Specifies to replace the old version when a new version is generated.

Permission tab

Specifies permissions of roles/users/groups on the Visual Analysis template. This tab is available when the Visual Analysis template is in the Public Reports folder.



Enable Setting Permissions

Enables the setting of permissions.

Available

Lists the roles/users/groups to which you can assign permissions. When organization is enabled, the Organization Name column will be available in the table.

- **Role**
  If checked, all roles will be displayed in the Available box for you to assign permissions.
- **User**
  If checked, all users will be displayed in the Available box for you to assign permissions.
- **Group**
  If checked, all groups will be displayed in the Available box for you to assign permissions.

Adds the selected role, user or group to the Selected box.

Removes the selected role, user or group from the Selected box.

Selected

Select a role/user/group in the Selected box and then select the permissions you would like the role/user/group to have on the Visual Analysis template. For what each permission is used for, see Permission.

When organization is enabled, the Organization Name column will be available in the table.

Page Report Studio dialogs

While using Page Report Studio, you will go through the following dialogs. This section introduces each of these dialogs in alphabetical order. You can go into the links for details.

- Add Aggregation dialog
- Aggregate On dialog
- Apply Style dialog
- Banded Object Properties dialog
- Banded Panel Properties dialog
- Banded Wizard
- Bottom N dialog
- Button Properties dialog
- Chart Definition dialog
- Chart Wizard
- Conditional Formatting dialog
- Crosstab Properties dialog
- Crosstab Wizard
- Customized Page dialog
- Data Field Properties dialog
- Edit Additional Values dialog
- Edit Aggregation dialog
- Edit Conditions dialog
- Encoding dialog
- Enter Values dialog
- Export dialog
- Filter Control Properties dialog
- Filter dialog
- Format Category (X) Axis dialog
- Format Label dialog
- Format Legend dialog
- Format Paper dialog
- Format Platform dialog
- Format Value (Y) Axis dialog
- Format Value (Y2) Axis dialog
- Formula Editor dialog
- Group Properties dialog
- Image Properties dialog
- Insert Aggregate dialog
- Insert Detail Column dialog
- Insert Filter Control dialog
- Insert Group Column dialog
- Insert Image dialog
- Insert Parameter Control dialog
- Insert Parameter Form Control dialog
- Insert Summary Column dialog
- Label Properties dialog
- Max Records dialog
- Multivalue Container Properties dialog
- Navigation Control Properties dialog
- New Page Report dialog
- New Report Tab dialog
- New Toolbar Name dialog
- Open Report Tabs dialog
- Options dialog
- Order/Select N dialog
- Page Properties dialog
- Parameter Control Properties dialog
- Parameter Field Properties dialog
- Parameter Form Control Properties dialog
- PDF Encrypt Option dialog
- PDF Sign Option dialog
- Printable Version dialog
- Query Filter dialog
- Rename Report Tab dialog
- Report Body Properties dialog
- Report Parameters dialog
- Report Tab Properties dialog
- Save As dialog
- Save Report Template dialog
- Search dialog
- Select Color dialog
- Select Field dialog
- Select Group Position dialog
- Select Resource dialog
- Select Values dialog
- Sort dialog
- Special Field Properties dialog
- Split dialog
- Table Cell Properties dialog
- Table Properties dialog
- Table Row Properties dialog
- Table Wizard
- Tabular Cell Properties dialog
- Tabular Properties dialog
- Text Box Properties dialog
- Top N dialog
- To Chart dialog
- To Crosstab dialog
- Turn to Page dialog
- Zoom dialog

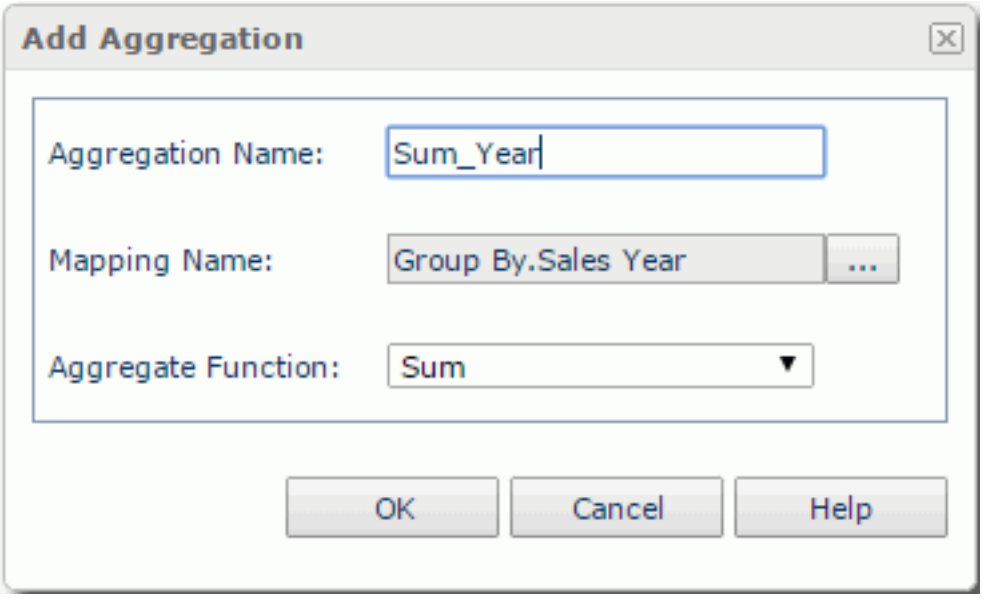
Note: The properties dialogs may be different for an advanced user and for a common user. An advanced user can modify some properties that a common user cannot, and can use formulas to control properties. The documents for properties dialogs in this section are based on advanced user.

To become an advanced user:

1. Log onto the JReport Administration page, click **Profile** on the system toolbar and select **Customize Profile** from the drop-down menu.
2. Select the Page Report Studio > Properties > **Advanced** tab and check the **Advanced User** option.
3. Click **OK** to accept the change.

Add Aggregation dialog

This dialog helps you to create a dynamic aggregation object in a report.



Aggregation Name

Specifies the display name of the aggregation object.

Mapping Name

Click ... to select a field or a formula on which the aggregation object is based in the Select Resource dialog. Then the field with its parent nodes path will be displayed.

Aggregate Function

Specifies the aggregate function of the aggregation object. The function will be applied to the group where the aggregation is placed or the entire report if the aggregation is placed in the report header or footer. For details about each function, see Math functions.

OK

Creates the aggregation object and closes the dialog.

Cancel

Cancels the creation and closes the dialog.

Help

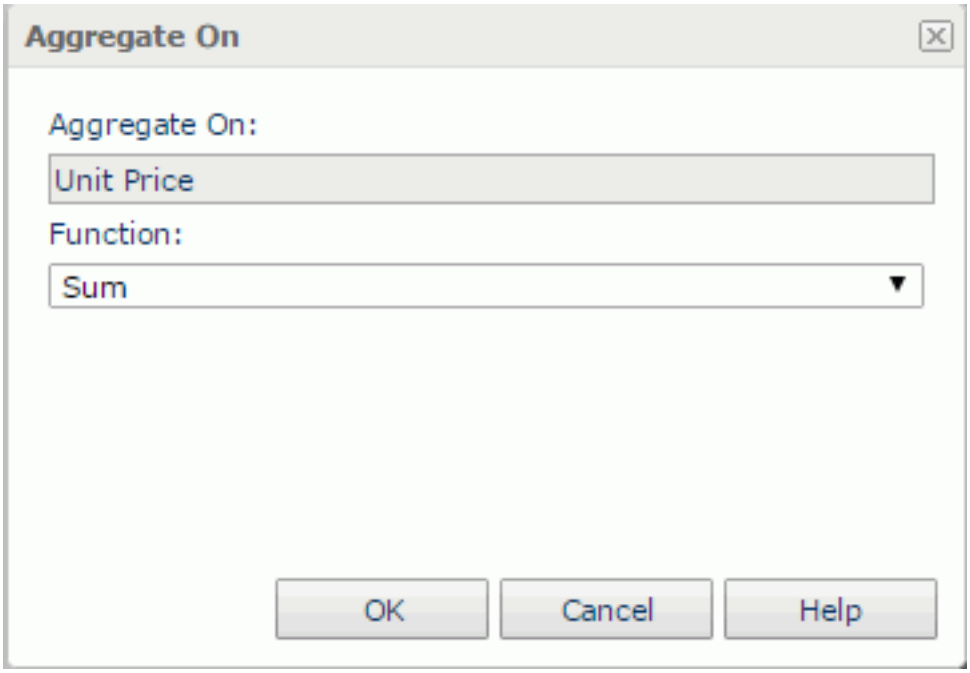
Displays the help document about this feature.

Related topics:

- Creating and using dynamic aggregation objects

Aggregate On dialog

The dialog helps you to summarize data in the selected table detail column.



Aggregate On

Displays the field bound with the detail column. This field is also the one on which the summary is based.

Function

Specifies the function to summarize the field in the detail column. For details about each function, see Math functions.

OK

Summarizes the detail column data with the specified function and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Help

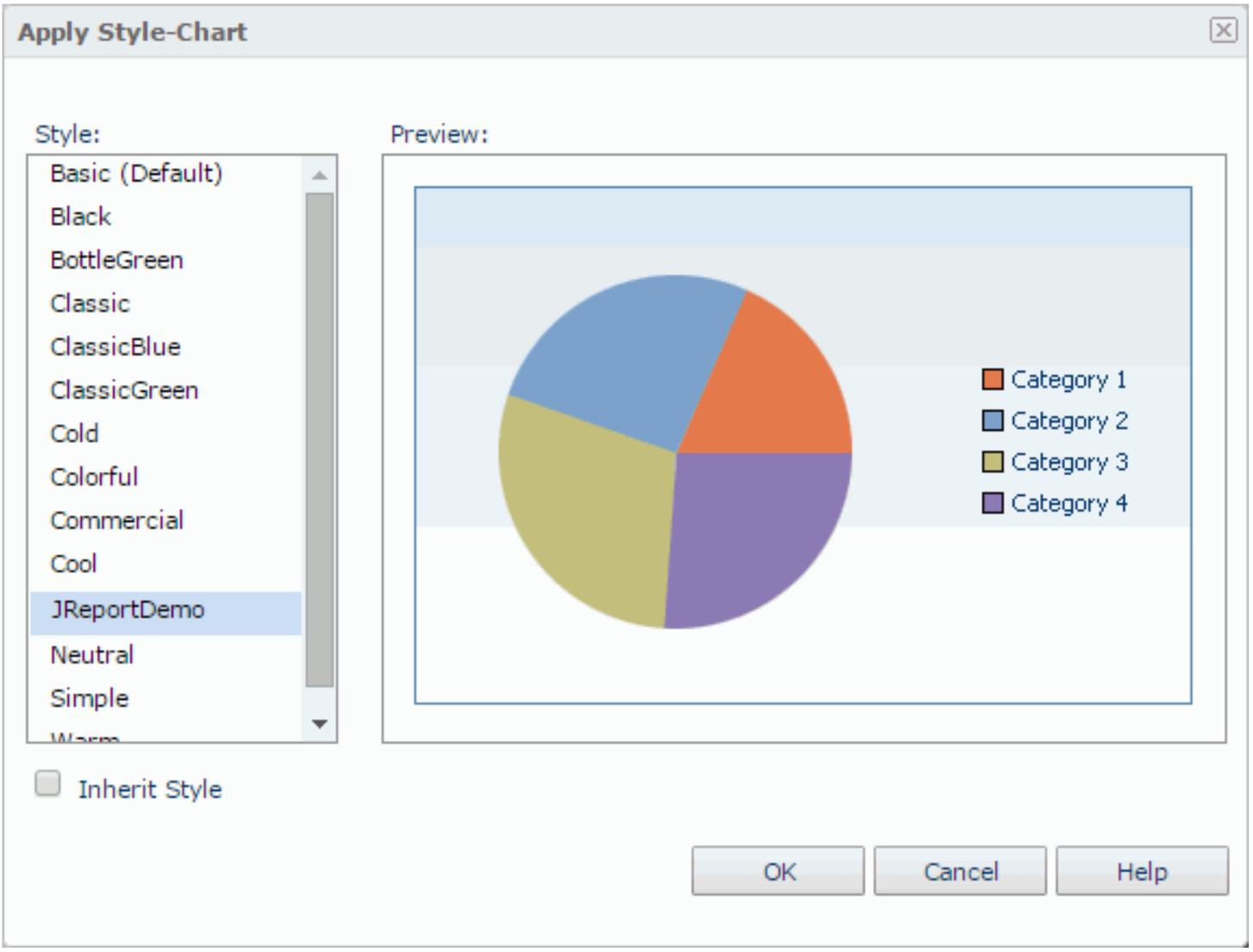
Displays the help document about this feature.

Related topics:

- Aggregating on a detail column

Apply Style dialog

This dialog helps you to set a style to the selected data component.



Style

Lists the styles applicable to the component.

- **Custom**
  There is no style information in this style and it is only used to handle reports created with previous versions which did not bind any style or the bound style cannot be found in the style list.

Preview

Displays a diagram illustrating the effect of the selected style on the component.

Inherit Style

Specifies to take the style of the parent component. The option is available only when the banded object, table, crosstab, or chart is contained in a banded object (for a chart, in a table also).

OK

Accepts all changes and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Help

Displays the help document about this feature.

Banded Object Properties dialog

This dialog helps you to edit the properties of a banded object. It contains the following tabs:

- General tab
- Border tab
- Others tab

OK

Applies the settings and closes this dialog.

Cancel

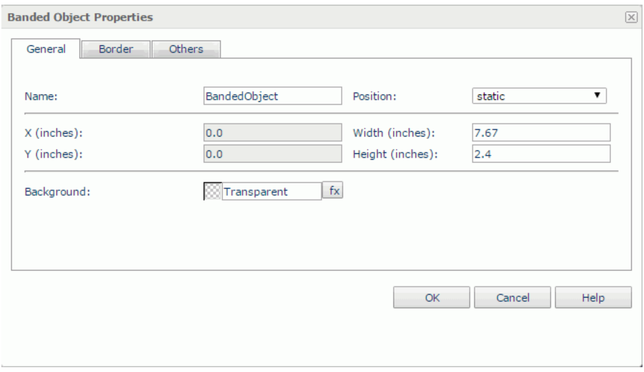
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the banded object.



Name

Specifies the display name of the banded object, which will be shown on the shortcut menu of the banded object.

Position

Displays the position mode of the banded object. If the banded object is directly contained in the report body, a tabular cell, or a text box, its position mode can be modified.

- **Absolute**: The banded object's position will be decided by its X and Y property values.
- **Static**: The banded object will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled.

X

Specifies the X coordinate of the banded object, in inches.

Y

Specifies the Y coordinate of the banded object, in inches.

Width

Specifies the width of the banded object, in inches.

Height

Specifies the height of the banded object, in inches.

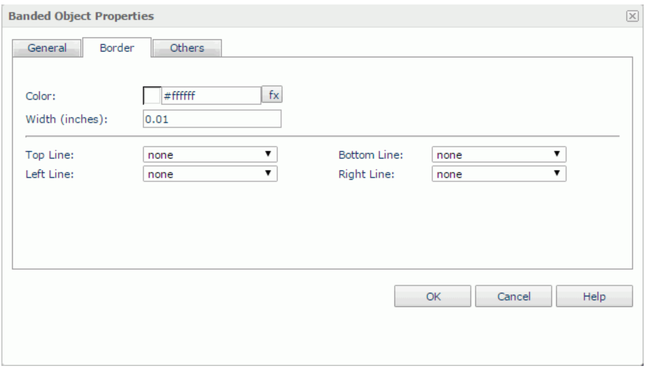
Background

Specifies the background color of the banded object.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the banded object.



Color

Specifies the border color. To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB.

Width

Specifies the border width in inches.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

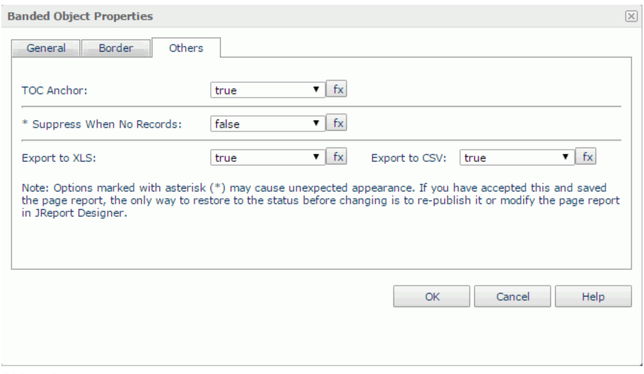
Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Others tab

You can use this tab to view and configure some miscellaneous settings.



TOC Anchor

Specifies whether or not to add the node that represents the banded object to the TOC tree that is displayed in the TOC Browser.

Suppress When No Records

If true and no records are returned to the banded object, the banded object will not be displayed.

Export to XLS

If true, the banded object will be exported when you save the report result as an XLS file (make sure to check Data Format in the Export dialog).

Export to CSV

If true, the banded object will be exported when you save the report result as a TXT file with Delimited Format selected.

Banded Panel Properties dialog

This dialog helps you to edit the properties of a banded panel. It contains the following tabs:

- General tab
- Border tab
- Others tab

OK

Applies the settings and closes this dialog.

Cancel

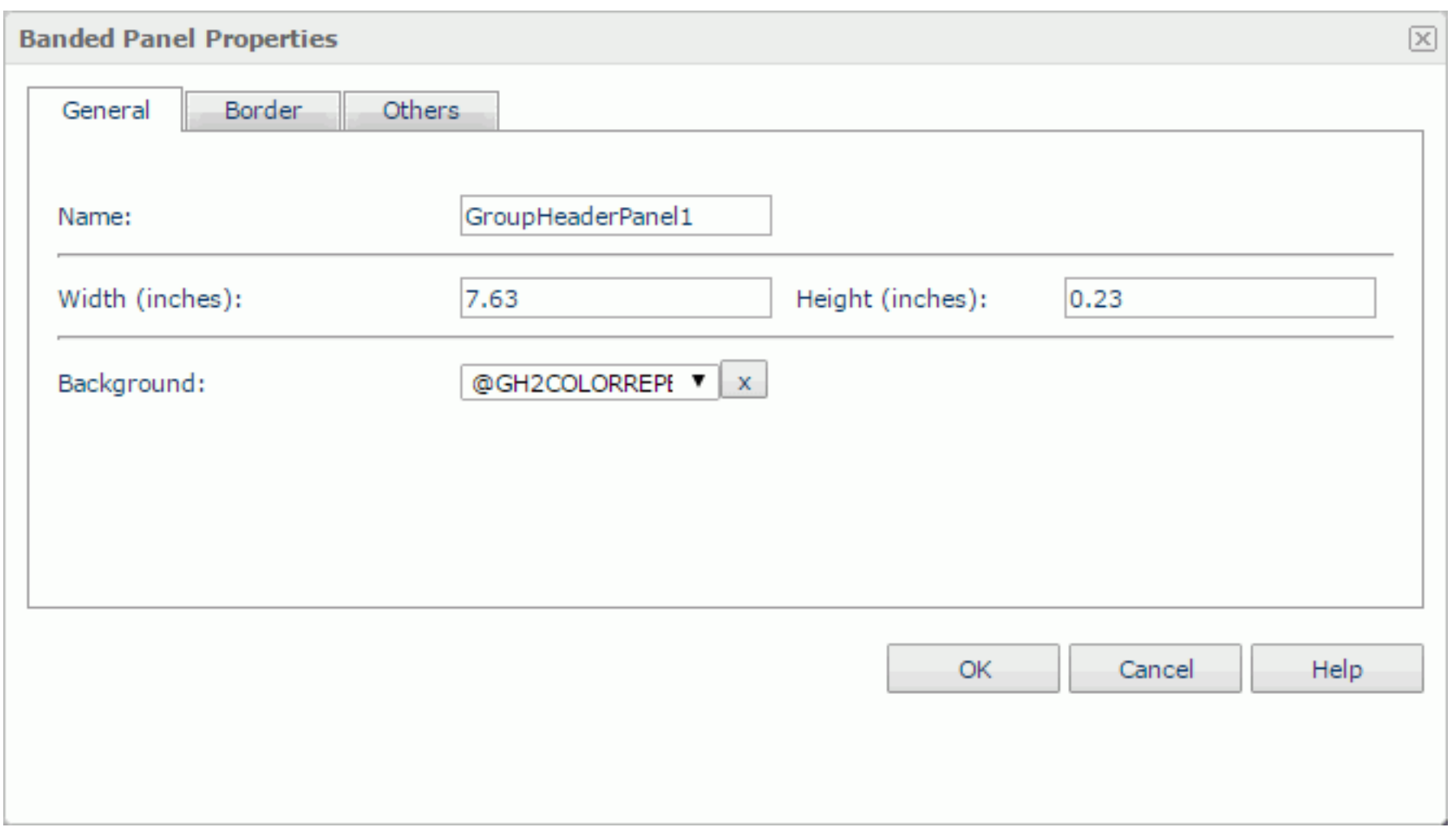
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the banded panel.



Name

Specifies the display name of the panel, which will be shown on the shortcut menu of the panel.

Width

Specifies the width of the panel in inches.

Height

Specifies the height of the panel in inches.

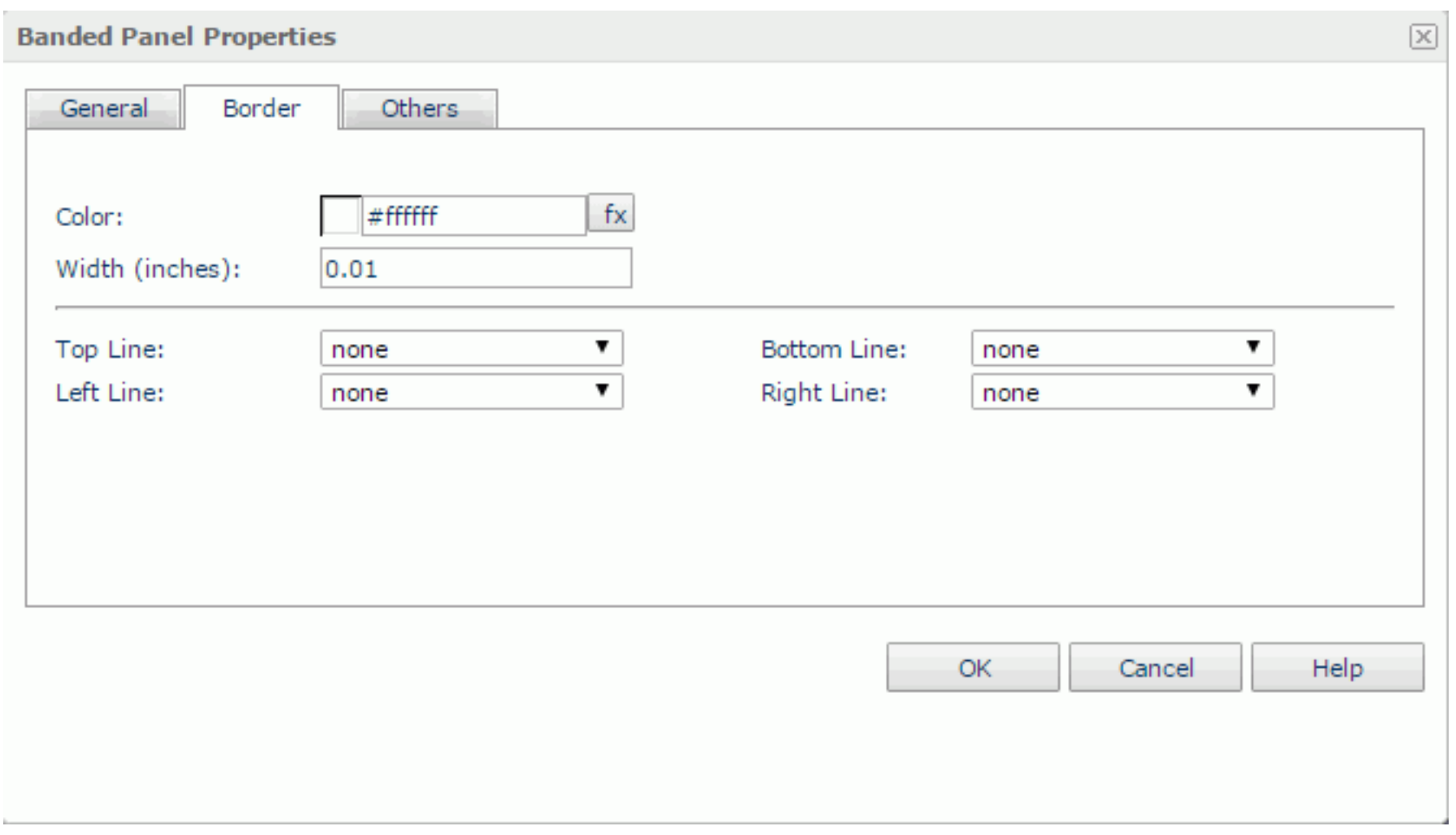
Background

Specifies the background color of the panel.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the banded panel.



Color

Specifies the border color. To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB.

Width

Specifies the border width in inches.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

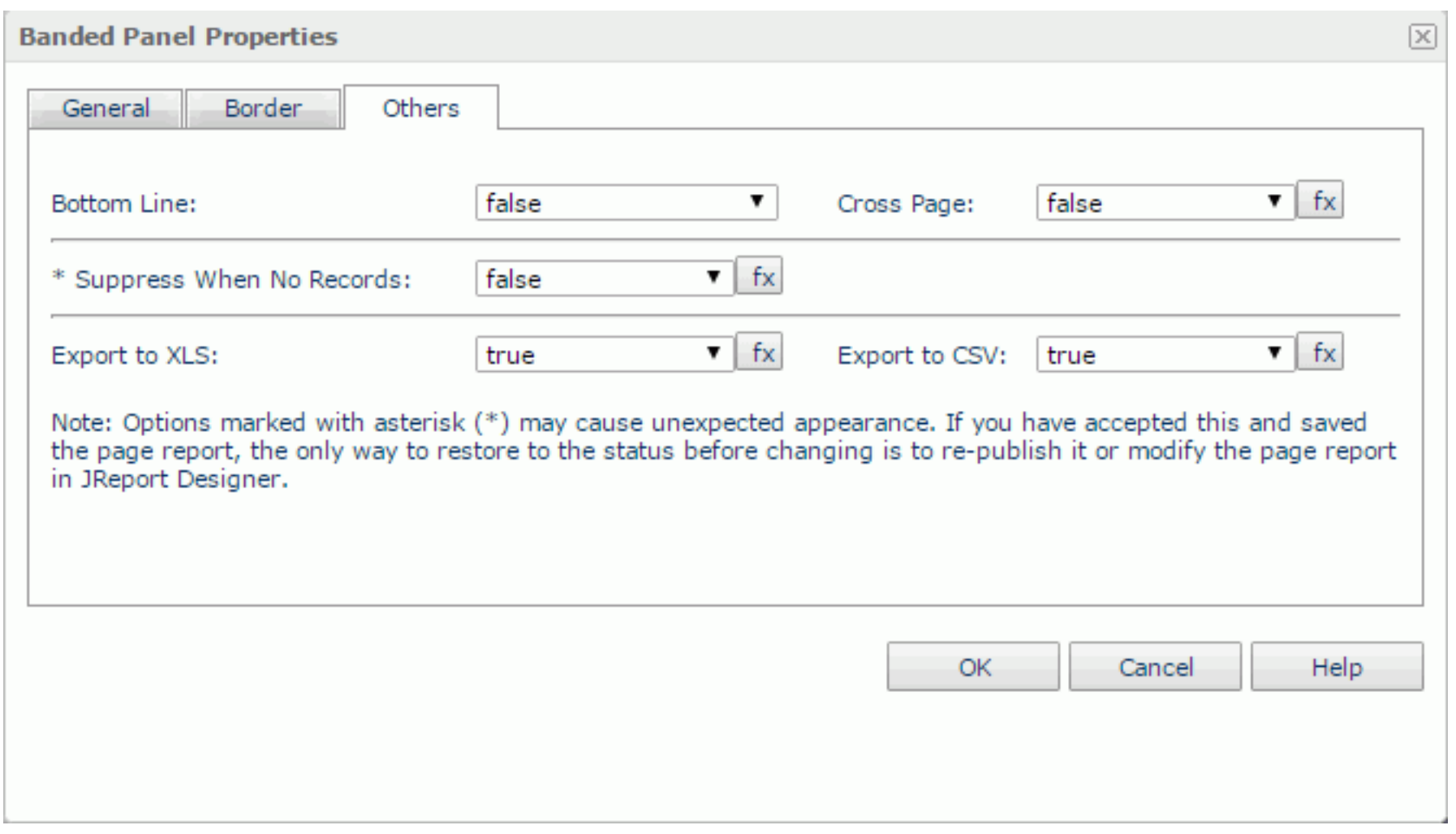
Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Others tab

You can use this tab to view and configure some miscellaneous settings.



Bottom Line

Specifies whether or not to show the bottom line of the panel.

Cross Page

When the current page cannot hold the whole panel, the panel will cross into the next page if the property is set to true, or else the whole panel will be displayed in the next page.

Suppress When No Records

If true and no records are returned to the panel, the panel will not be displayed.

Export to XLS

If true, the panel will be exported when you save the report result as an XLS file (make sure to check Data Format in the Export dialog).

Export to CSV

If true, the panel will be exported when you save the report result as a TXT file with Delimited Format selected.

Banded Wizard

This wizard guides you through the process of creating a banded object. It contains the following screens:

- Data screen
- Display screen
- Group screen
- Summary screen
- Query Filter screen
- Style screen

Back
Returns to the previous screen.

Next
Goes to the next screen.

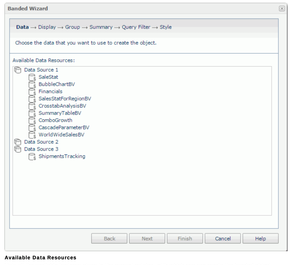
Finish
Creates the banded object and closes the wizard.

Cancel
Closes the wizard without creating a banded object.

Help
Displays the help document about this feature.

Data screen

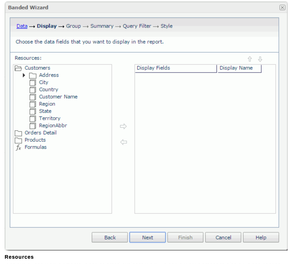
Specifies the business view to use to create the banded object. This screen is hidden when there is only one business view in the current catalog.



Available Data Resources
Lists all the available business views in the current catalog, with which you can create the banded object.

Display screen

Specifies the fields to be displayed in the banded object.



Resources
Displays all the view elements in the selected business view. To add a view element to display in the banded object, select it and then click .

Display Fields
Lists the view elements you have selected to display in the banded object. For an added view element which you do not want, select it and then click .

Display Name
Specifies the display names of the added view elements.
The downward order of the added elements determines the order of them displayed in the banded object from left to right. You can use the following two arrows to adjust the order.

Moves the selected view element one step up.

Moves the selected view element one step down.

Group screen

Specifies the fields to group the data in the banded object.

Resources
Displays all the available group objects you can use to group the data in the banded object. To add a group object as a group field, select it and then click .

Group By
Lists all the group objects that have been added as the group fields. To cancel a group from being a group field, select it and then click .

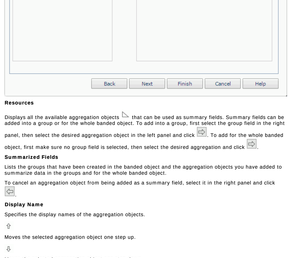
Sort
Specifies the sort order for each group: Ascend, Descend, or No Sort.
The downward order of the added group fields determines the order of the groups in the banded object from first to last. You can use the following two arrows to adjust the order.

Moves the selected group one step up.

Moves the selected group one step down.

Summary screen

Specifies summary fields that calculate data based on a group or on the whole banded object.



Resources
Displays all the available aggregation objects that can be used as summary fields. Summary fields can be added into a group or for the whole banded object. To add into a group, first select the group field in the right panel, then select the desired aggregation object in the left panel and click . To add for the whole banded object, first make sure no group field is selected, then select the desired aggregation and click .

Summarized Fields
Lists the groups that have been created in the banded object and the aggregation objects you have added to summarize data in the groups and for the whole banded object.
To cancel an aggregation object from being added as a summary field, select it in the right panel and click .

Display Name
Specifies the display names of the aggregation objects.

Moves the selected aggregation object one step up.

Moves the selected aggregation object one step down.

Query Filter screen

Specifies the filter which you want to apply to the selected business view.

In this screen, all the predefined filters of the business view are listed in the Query Filter drop-down list. You can choose one of them to apply. If you prefer to define a filter on your own, select **User Defined** from the drop-down list, and then define it according to your requirements.
For details about options in the screen, refer to Query Filter dialog.

Style screen

Specifies the style of the banded object. This screen is hidden when there is only one style available to be applied to the banded object.

Style
Lists all the available styles for you to select from.

Inherit Style
Specifies to take the style of the parent component. The option is available only when you specify to insert the banded object into another banded object.

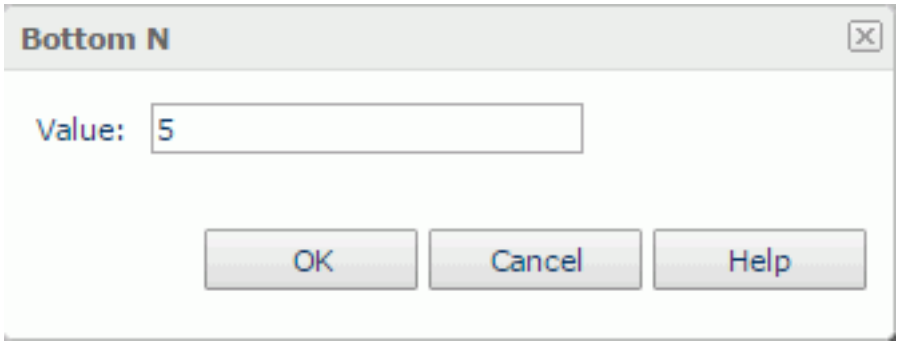
Preview
Shows a preview of the banded object in the selected style.

Related topics:

- Creating a banded report

Bottom N dialog

This dialog helps you to filter data to display records that meet the Bottom N condition.



Value

Enter a positive integer number N here so that records with the field value equal to one of the bottom N field values will be displayed.

OK

Applies the settings and closes the dialog.

Cancel

Closes the dialog and discards any changes.

Help

Displays the help document about this feature.

Related topics:

- Filtering the data components in a report

Button Properties dialog

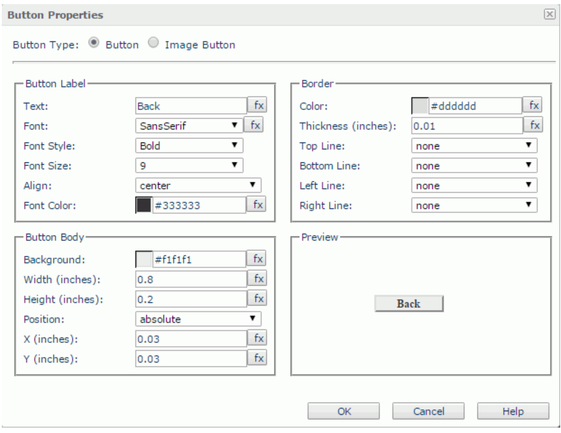
This dialog helps you to modify the properties of a button in a navigation control.

Button Type

- Button
  The button is displayed as a normal button.
- Image Button
  The button is displayed as an image.

Button

When Button is selected as the button type, the following options are available.



Button Label

Specifies the properties of the button label.

- **Text**
  Specifies the text of the label.
- **Font**
  Specifies the font face of the text.
- **Font Style**
  Specifies the font style of the text.
- **Font Size**
  Specifies the font size of the text.
- **Align**
  Specifies the alignment way of the label in the button.
- **Font Color**
  Specifies the font color of the text.

Border

Specifies the properties of the button border.

- **Color**
  Specifies the border color.
- **Thickness**
  Specifies the border width in inches.
- **Top Line**
  Specifies the style of the top border line.
- **Bottom Line**
  Specifies the style of the bottom border line.
- **Left Line**
  Specifies the style of the left border line.
- **Right Line**
  Specifies the style of the right border line.

Button Body

Specifies the properties of the button body.

- **Background**
  Specifies the background color of the button body.

  To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.
- **Width**
  Specifies the width of the button in inches.
- **Height**
  Specifies the height of the button in inches.
- **Position**
  Displays the position mode of the button. If the button is directly contained in the report body, a tabular cell, or a text box, its position mode can be modified.
  - **Absolute**: The button's position will be decided by its X and Y property values.
  - **Static**: The button will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled.
- **X**
  Specifies the X coordinate of the button, in inches.
- **Y**
  Specifies the Y coordinate of the button, in inches.

Preview

Displays a preview of the button according to the button properties.

Image Button

When Button Image is selected as the button type, the following options are available.

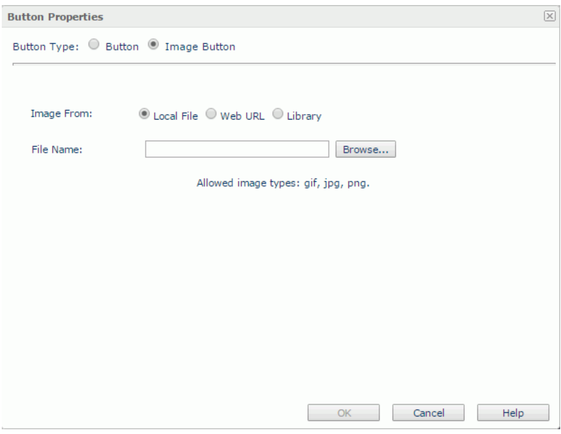


Image From

Specifies the source of the image file.

- **Local File**
  Specifies to use an image from the local file system. The types and size of the images that can be used are defined by the administrator in the Upload panel.
  - **File Name**
    Specifies the path and name of the image file. You can click the Browse button to locate the image file.
- **Web URL**
  Specifies to use an image via URL.
  - **Image URL**
    Specifies the URL of the image file. JReport will record the latest 10 entered URLs in the drop-down list.

    Note: If your JReport Server is in an intranet, to successfully access the image via URL, you need to add the parameters `-Dhttp.proxyHost=XXX -Dhttp.proxyPort=XX` to the server's startup file JRServer.bat, which locates in `<install_root>\bin`.
- **Library**
  Specifies to use an existing image.
  - **My Images**
    The My Images folder is a virtual location where JReport Server stores the images that have once been inserted into reports. Select the one you want to use.
  - **Preview**
    Displays a preview of the selected image.

OK

Applies the button properties and closes this dialog.

Cancel

Cancels the changes and closes this dialog.

Help

Displays the help document about this feature.

Chart Definition dialog

The dialog helps you to modify the definition of a chart. It contains the following tabs:

- Chart Type tab
- Display tab
- Style tab

OK

Applies the settings and closes the dialog.

Cancel

Cancels the settings and closes the dialog.

Help

Displays the help document about this feature.

Chart Type tab

This tab allows you to set the type of the chart.

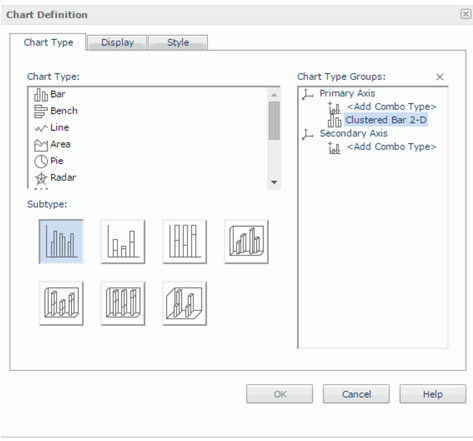


Chart Type

Lists all the chart types.

Subtype

Shows subtypes of the selected chart type in thumbnail form.

Chart Type Groups

Lists the subtypes defined for the chart.

To replace a chart type with another one, first select it, then select the wanted chart type and a sub type.

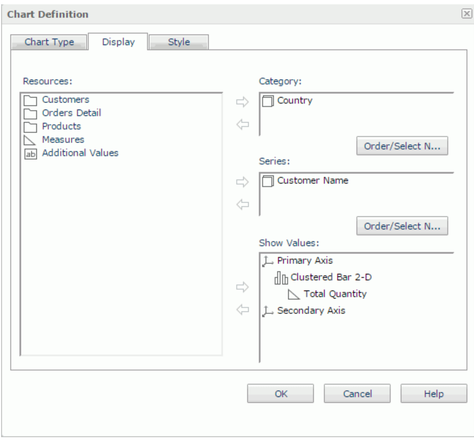
If you want to create a combo chart, click **<Add Combo Type>** of Primary Axis or Secondary Axis in the Chart Type Groups box, and an additional subtype will be added. To replace the additional subtype, select it, then specify the required type and subtype respectively. To add more subtypes, repeat the procedures.

×

Removes the selected subtype. At least one type should remains for the Primary Axis so as to create the chart.

Display tab

This tab allows you to set the fields that will be displayed in the chart.



Resources

Displays all the group objects and aggregation objects used in the chart.

Adds the selected view element to the chart.

Removes the selected view element.

Category

Lists the group object that will be displayed on the category axis of the chart.

Series

Lists the group object that will be displayed on the series axis of the chart.

Show Values

Lists the aggregation objects and additional values that will be displayed on the value axis of the chart.

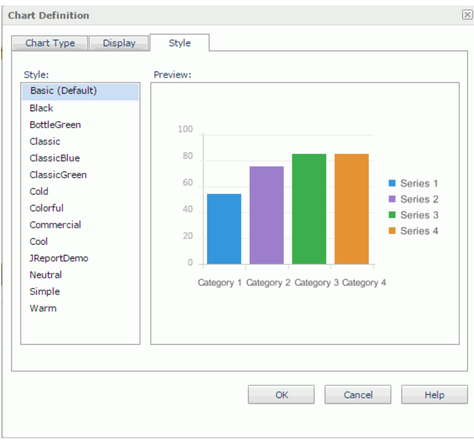
- Opens the Edit Additional Value dialog to edit an additional value. Available only when an additional value is selected in the Show Values box.

Order/Select N

Opens the Order/Select N dialog to define the sort order and Select N condition in the chart.

Style tab

This tab allows you to select a style for the chart. It is hidden when there is only one style available.



Style

Lists the available styles.

- **Custom**
  There is no style information in this style and it is only used to support reports built with previous versions which did not bind any style or the bound style cannot be found in the style list.

Preview

Displays a diagram illustrating the effect of the selected style on the chart.

Related topics:

- Modifying the definition of a chart

Chart Wizard

This wizard guides you through the process of creating a chart. It contains the following screens:

- Data screen
- Type screen
- Display screen
- Query Filter screen
- Style screen

Back

Returns to the previous screen.

Next

Goes to the next screen.

Finish

Creates the chart and closes the wizard.

Cancel

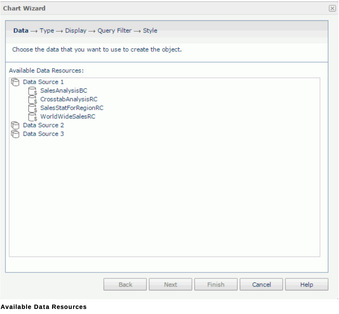
Closes the wizard without creating a chart.

Help

Displays the help document about this feature.

Data screen

Specifies the business view to use to create the chart. This screen is hidden when there is only one business view in the current catalog.



Available Data Resources

Lists all the available business views in the current catalog, with which you can create the chart.

Type screen

Specifies the type of the chart.

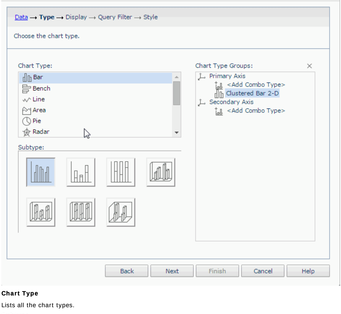


Chart Type

Lists all the chart types.

Subtype

Shows subtypes of the selected chart type in thumbnail form.

Chart Type Groups

Lists the subtypes defined for the chart.

A default chart type exists in the Chart Type Groups box. To replace it with another one, first select it, then select the wanted chart type and a sub type.

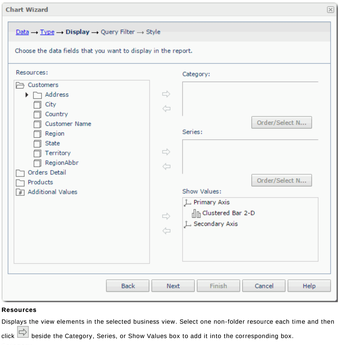
If you want to create a combo chart, click **<Add Combo Type>** of Primary Axis or Secondary Axis in the Chart Type Groups box, and an additional subtype will be added. To replace the additional subtype, select it, then specify the required type and subtype respectively. To add more subtypes, repeat the procedures.

×

Removes the selected subtype. At least one type should remain for the Primary Axis so as to create the chart.

Display screen

Specifies the fields to be displayed in the chart.



Resources

Displays the view elements in the selected business view. Select one non-folder resource each time and then click beside the Category, Series, or Show Values box to add it into the corresponding box.

Category

Lists the group object that will be displayed on the category axis of the chart. If you don't want the current resource, select it and click on the left to remove it.

Series

Lists the group object that will be displayed on the series axis of the chart. If you don't want the current resource, select it and click on the left to remove it.

Show Values

Lists the aggregation objects and additional values that will be displayed on the value axis of the chart. For a combo chart, specify resources for each chart type. To add a resource to a chart type, first select the resource and the chart type separately, then click .

To remove a resource from the box, select it and click on the left.

- Opens the Edit Additional Value dialog to edit an additional value. Available only when an additional value is selected in the Show Values box.

Order/Select N

Opens the Order/Select N dialog to define the sort order and Select N condition in the chart.

Query Filter screen

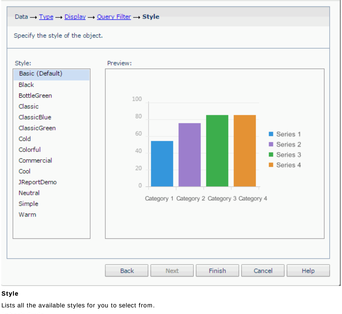
Specifies the filter which you want to apply to the selected business view.

In this screen, all the predefined filters of the business view are listed in the Query Filter drop-down list. You can choose one of them to apply. If you prefer to define a filter on your own, select **User Defined** from the drop-down list, and then define it according to your requirements.

For details about options in the screen, refer to Query Filter dialog.

Style screen

Specifies the style of the chart. This screen is hidden when there is only one style available to be applied to the chart.



Style

Lists all the available styles for you to select from.

Inherit Style

Specifies to take the style of the parent component. The option is available only when you specify to insert the chart into a banded object or table.

Preview

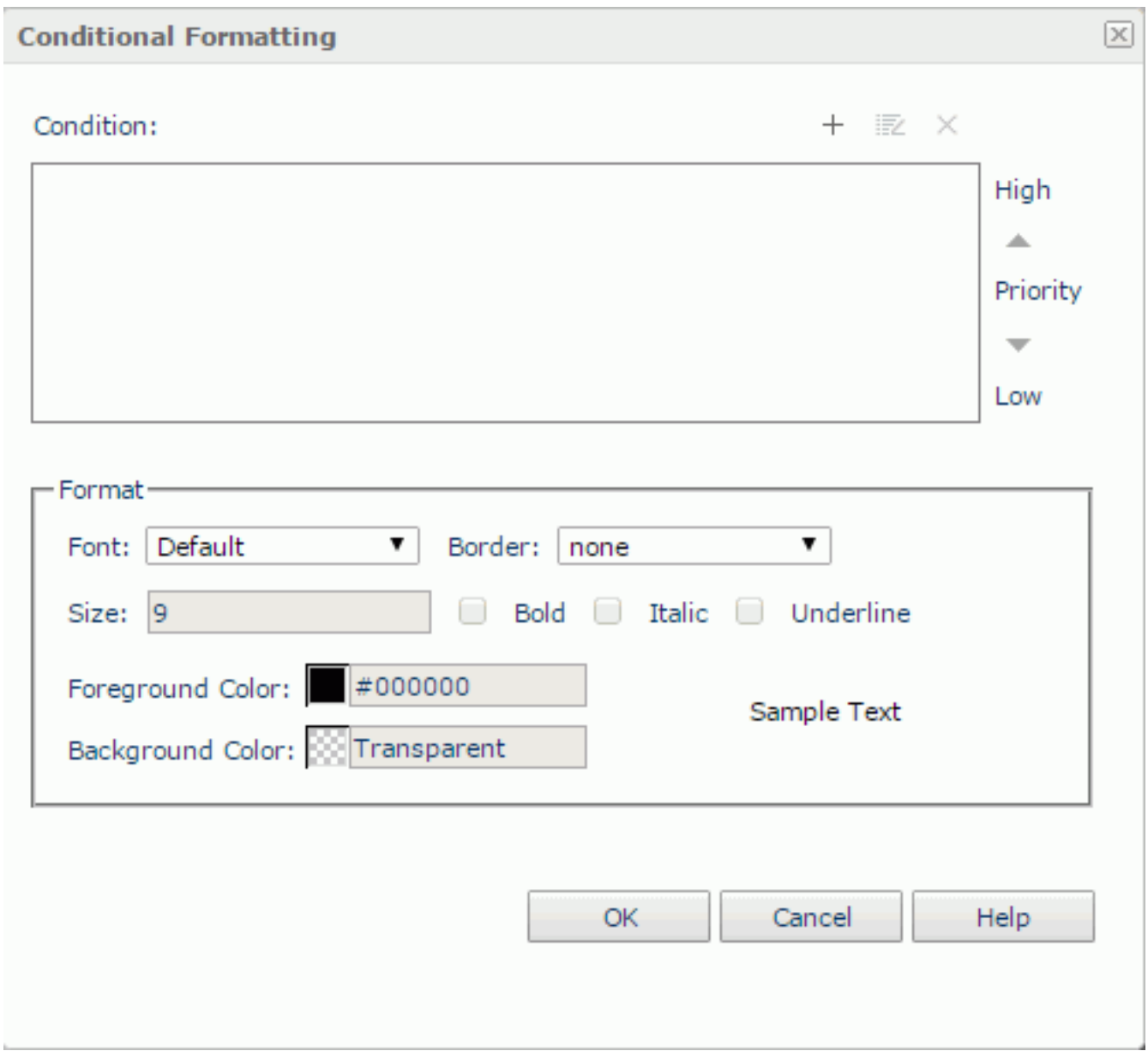
Shows a preview of the selected style.

Related topics:

- Creating a chart report

Conditional Formatting dialog

This dialog helps you to add some conditional formats to values of the selected field.



Condition

Displays all the conditions created on the field.

- Creates a new condition using the Edit Conditions dialog.
- Edits the selected condition using the Edit Conditions dialog.
- Removes the selected condition.
- **Priority**
  Specifies the priority of each condition.
  - Moves a condition up for a higher priority.
  - Moves a condition down for a lower priority.

Format

You can specify a format for each condition. The format will be applied to the field values when the specified condition is fulfilled.

- **Font**
  Specifies the font type for the field values.
- **Border**
  Specifies the border line style for the field values.
- **Size**
  Specifies the font size for the field values.
- **Bold**
  Specifies whether or not to bold the field values.
- **Italic**
  Specifies whether or not to make the field values italic.
- **Underline**
  Specifies whether or not to underline the field values.
- **Foreground Color**
  Specifies the foreground color for the field values.
- **Background Color**
  Specifies the background color for the field values.
- **Sample Text**
  Displays a preview sample of your settings.

OK

Applies the settings and exits the dialog.

Cancel

Cancels the settings and closes the dialog.

Help

Displays the help document about this feature.

Related topics:

- Adding conditional formats to fields

Crosstab Properties dialog

This dialog helps you to edit the properties of a crosstab. It contains the following tabs:

- General tab
- Border tab
- Crosstab tab
- Others tab

OK

Applies the settings and closes this dialog.

Cancel

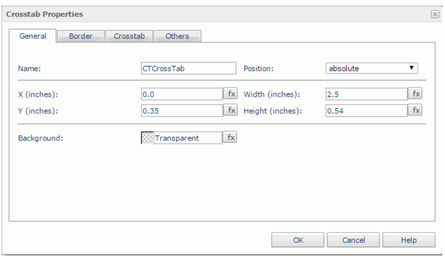
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the crosstab.



Name

Specifies the display name of the crosstab, which will be shown on the shortcut menu of the crosstab.

Position

Displays the position mode of the crosstab. If the crosstab is directly contained in the report body, a tabular cell, or a text box, its position mode can be modified.

- **Absolute**: The crosstab's position will be decided by its X and Y property values.
- **Static**: The crosstab will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled.

X

Specifies the X coordinate of the crosstab, in inches.

Y

Specifies the Y coordinate of the crosstab, in inches.

Width

Specifies the width of the crosstab, in inches.

Height

Specifies the height of the crosstab, in inches.

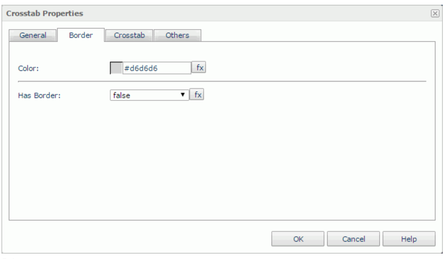
Background

Specifies the background color of the crosstab.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the crosstab.



Color

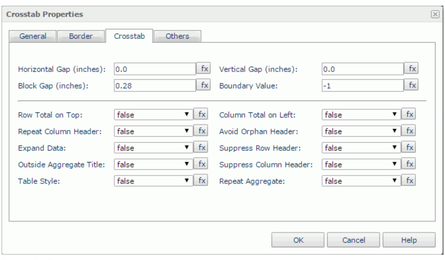
Specifies the color of the borders. To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the border transparent, input Transparent in the text box.

Has Border

Specifies whether or not to show the borders.

Crosstab tab

This tab shows the layout-related information of the crosstab.



Horizontal Gap

Specifies the space between the content and left/right edge of a crosstab cell.

Vertical Gap

Specifies the space between the content and top/bottom edge of a crosstab cell.

Block Gap

Specifies the spacing between each part of the crosstab if the crosstab will be split into more than one part.

Boundary Value

Specifies the number of columns in one aggregate cell when the crosstab is displayed horizontally, or rows when displayed vertically.

Row Total on Top

Specifies whether or not to display the Total row in the first row of the crosstab.

Column Total on Left

Specifies whether or not to display the Total column in the first column in the crosstab.

Repeat Column Header

Specifies whether or not to repeat column headings on every page.

Avoid Orphan Header

Sometimes the column header happens to be at the bottom of a page. To keep the column header together with the data in the next page, set this property to true.

Expand Data

Specifies whether or not to enable Page Report Studio users to expand or collapse dimensions in the crosstab.

Suppress Row Header

Specifies whether or not to hide the row headers.

Outside Aggregate Title

Specifies whether or not to place the titles of aggregate fields outside.

Suppress Column Header

Specifies whether or not to hide the column headers.

Table Style

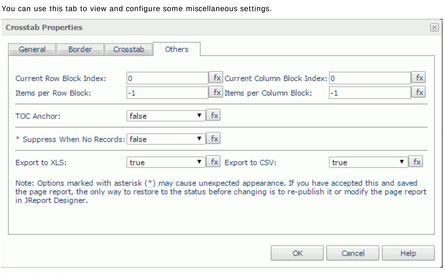
Specifies whether or not to add headers to the Total rows and columns.

Repeat Aggregate

Specifies whether or not to repeat the crosstab for different aggregate fields.

Others tab

You can use this tab to view and configure some miscellaneous settings.



Current Row Block Index

Specifies the vertical index of the data block that will be displayed. 0 means the first block index, and 1 the second, and so on.

Current Column Block Index

Specifies the horizontal index of the data block that will be displayed. 0 means the first block index, and 1 the second, and so on.

Items per Row Block

Specifies the number of rows of records in each data block.

Items per Column Block

Specifies the number of columns of records in each data block.

The four properties work together to control the data of the crosstab to be displayed in continuous page mode: Current Row Block Index, Current Column Block Index, Items per Row Block, and Items per Column Block.

TOC Anchor

Specifies whether or not to add the node that represents the crosstab to the TOC tree that is displayed in the TOC Browser.

Suppress When No Records

If true and no records are returned by the report, the crosstab will not be displayed.

Export to XLS

If true, the crosstab will be exported when you save the report result as an XLS file (make sure to check Data Format in the Export dialog).

Export to CSV

If true, the crosstab will be exported when you save the report result as a TXT file with Delimited Format selected.

Crosstab Wizard

This wizard guides you through the process of creating a crosstab. It contains the following screens:

- Data screen
- Display screen
- Query Filter screen
- Style screen

Back

Returns to the previous tab.

Next

Goes to the next tab.

Finish

Creates the crosstab and closes the wizard.

Cancel

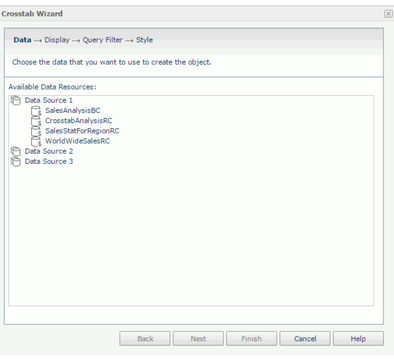
Closes the wizard without creating a crosstab.

Help

Displays the help document about this feature.

Data screen

Specifies the business view to use to create the crosstab. This screen is hidden when there is only one business view in the current catalog.

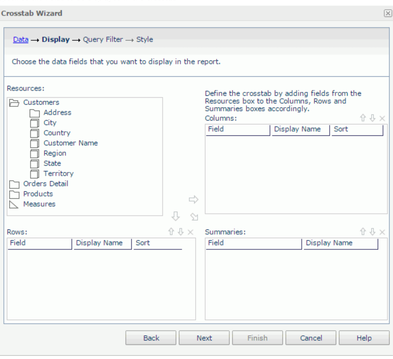


Available Data Resources

Lists all the available business views in the current catalog, with which you can create the crosstab.

Display screen

Specifies the fields to be displayed in the crosstab.



Resources

Displays the view elements in the selected business view. Select one non-folder resource each time and then click a proper arrow button to add it into the corresponding box.

Adds the selected group object to be displayed on the columns of the crosstab.

Adds the selected group object to be displayed on the rows of the crosstab.

Adds the selected aggregation object to be the aggregate field of the crosstab.

Columns/Rows

- **Field**
  Lists the group objects that will be displayed on the columns/rows of the crosstab.
- **Display Name**
  Specifies the display names of the group objects. By default these are blank and no labels will be created for the group objects to name the columns/rows. You can click the cells to edit them if required.
- **Sort**
  Specifies the sort order of the group objects.

Summaries

- **Field**
  Lists the aggregation objects that will be the aggregate fields of the crosstab.
- **Display Name**
  Specifies the display names of the aggregation objects. By default these are blank and no labels will be created for the aggregation objects to name the summaries. You can click the cells to edit them if required.

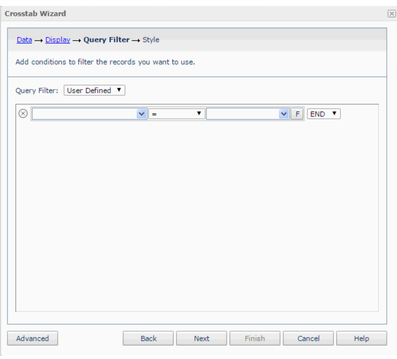
Moves the selected resource one level up. The order here defines the order of the resources displayed in the crosstab from left to right or from top to bottom.

Moves the selected resource one level down. The order here defines the order of the resources displayed in the crosstab from left to right or from top to bottom.

Removes the selected resource from the crosstab.

Query Filter screen

Specifies the filter which you want to apply to the selected business view.

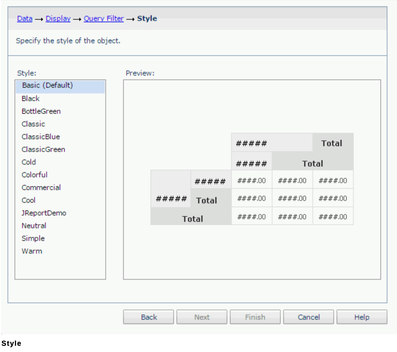


In this screen, all the predefined filters of the business view are listed in the Query Filter drop-down list. You can choose one of them to apply. If you prefer to define a filter on your own, select **User Defined** from the drop-down list, and then define it according to your requirements.

For details about options in the screen, refer to Query Filter dialog.

Style screen

Specifies the style of the crosstab. This screen is hidden when there is only one style available to be applied to the crosstab.



Style

Lists all the available styles for you to select from.

Inherit Style

Specifies to take the style of the parent component. The option is available only when you specify to insert the crosstab into a banded object.

Preview

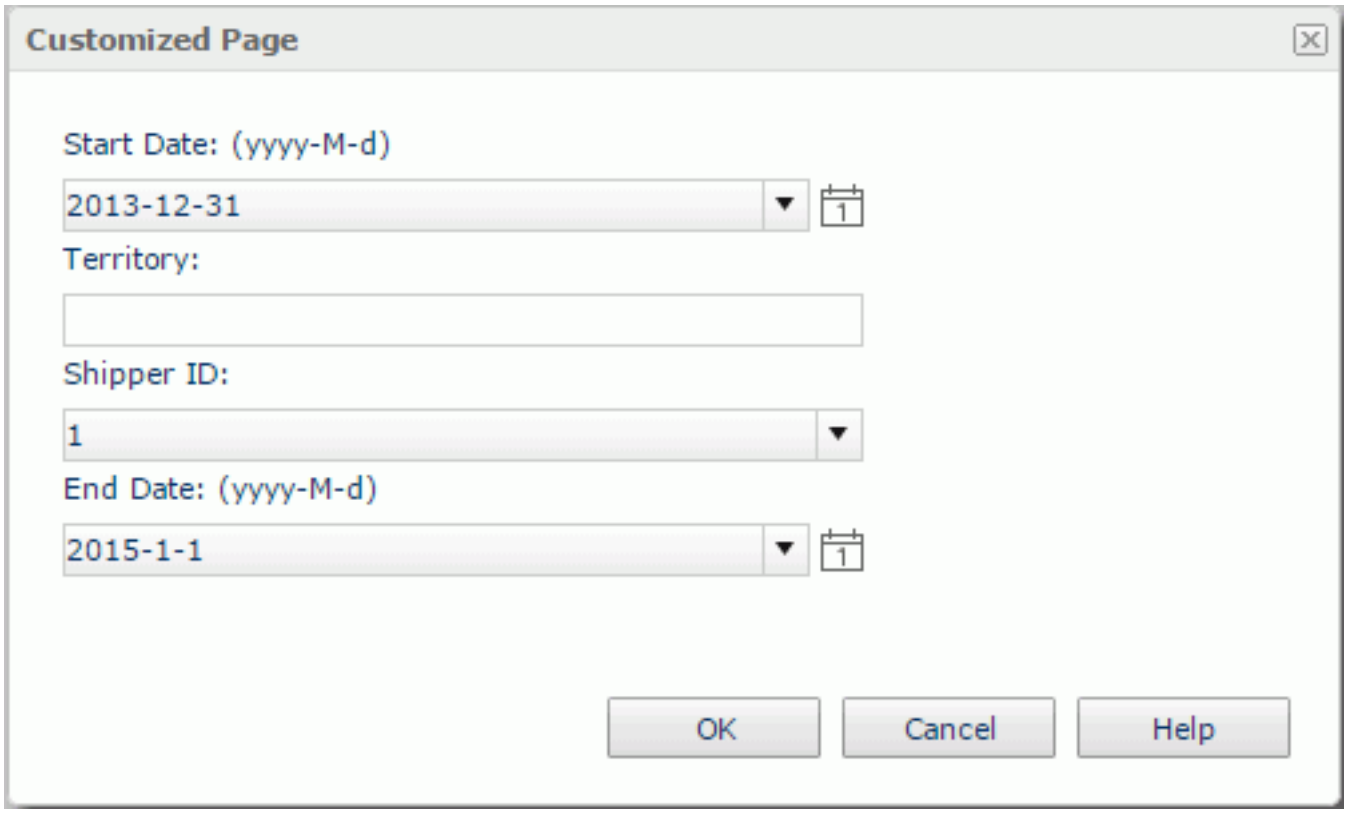
Shows a preview of the selected style.

Related topics:

- Creating a crosstab report

Customized Page dialog

This dialog appears when the web action user_popCustomizedPage() bound with a web control is triggered. It by default shows all the parameters the current report and its subreport use. You can change the parameter values to re-run the report according to your requirement. Also, you can customize the JSP page of this dialog to make it display what you need by using the Page Report Studio API.



OK

Applies the changes and closes the dialog.

Cancel

Cancels the changes and exits the dialog.

Help

Displays the help document about this feature.

Data Field Properties dialog

Use the Data Field Properties dialog to define or modify a data field. It contains the following tabs:

- General
- Format
- Data Field
- Value List
- Value Set
- Value List

Cancel

Cancels the settings and closes the dialog.

Save

Saves the settings and closes the dialog.

Help

Displays the Help document about this dialog.

General

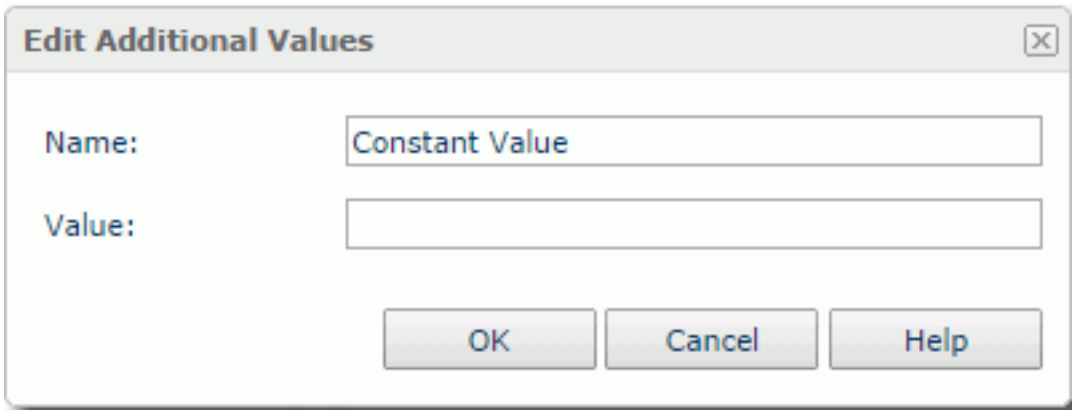
Use this tab to define the general characteristics of the data field.

Data Field Properties
General
Name
Data Type
Text Value
Text
Number
Date
Boolean
Length
Decimal Digits
Input Mask
Mandatory
Default Value
Input Mask Type
Mask Character
Input Mask Value
Help Text
Tooltip Text
Help Text
Tooltip Text
Help Text
Help Text
Help

Edit Additional Values dialog

This dialog helps you to edit an additional value. It varies according to value types.

If it is a constant value, options in the dialog are as follows.



Name

Specifies the display name of the constant value, which will be shown under the subtype in the Show Values box.

Value

Specifies a float number for the constant value.

OK

Applies the settings and closes the dialog.

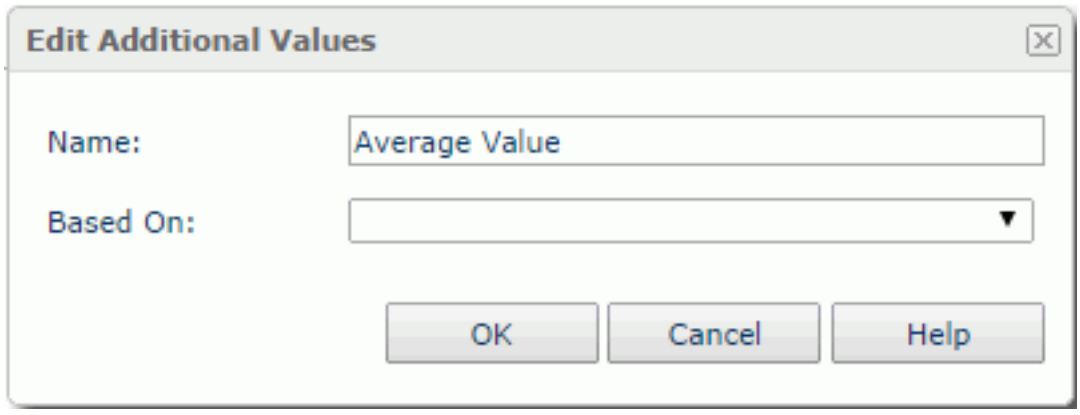
Cancel

Closes the dialog and discards any changes.

Help

Displays the help document about this feature.

If it is an average value, options in the dialog are as follows.



Name

Specifies the display name of the average value, which will be shown under the subtype in the Show Values box.

Based On

Specifies the aggregation object based on which the average value will be calculated.

OK

Applies the settings and closes the dialog.

Cancel

Closes the dialog and discards any changes.

Help

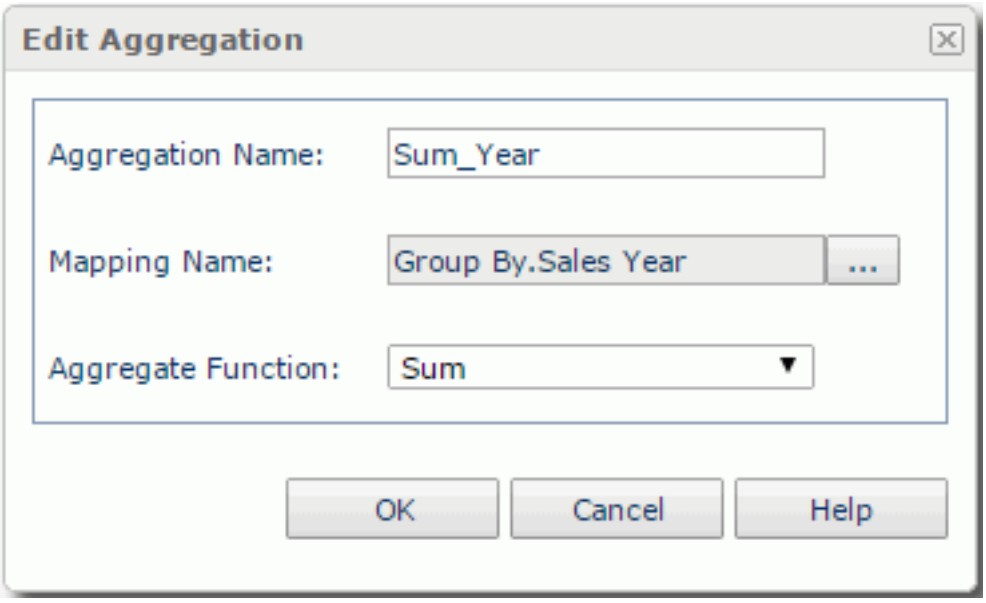
Displays the help document about this feature.

Related topics:

- Creating a chart report

Edit Aggregation dialog

This dialog helps you to edit an existing aggregation object.



Aggregation Name

Specifies the display name of the aggregation object.

Mapping Name

Specifies the mapping name of the field to which the aggregation object is mapped. Click ... to select the field in the Select Resource dialog.

This option and the button are disabled when you specify to create an aggregation object on a dynamic formula.

Aggregate Function

Specifies the aggregate function of the aggregation object. The function will be applied according to the group where the aggregation is placed or the entire report if placed in the report header or footer. For details about each function, see Math functions.

OK

Applies the changes to the aggregation object and closes the dialog.

Cancel

Cancels the operation and closes the dialog.

Help

Displays the help document about this feature.

Related topics:

- Creating and using dynamic aggregation objects

Edit Conditions dialog

This dialog helps you to edit a condition for the selected field for adding conditional format. It contains the following two modes:

- Basic
- Advanced

Advanced / Basic

Switches the dialog to the advanced/basic mode.

OK

Applies the settings and exits the dialog.

Cancel

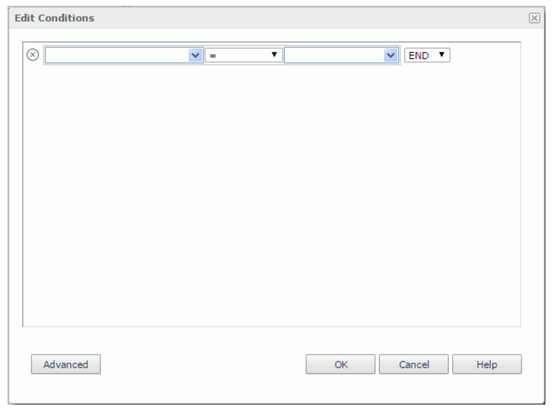
Cancels the settings and closes the dialog.

Help

Displays the help document about this feature.

Basic mode

The basic mode provides function for creating simple filter conditions which are connected by the AND and OR operators.



Deletes the current condition line. Each condition line contains an expression with a logic operator which either connects the current expression with the following one or ends when it is the last line. An expression is composed of a field, an operator, and a value.

Field

Specifies the field to be filtered.

Operator

Specifies the operator to compose the filter expression.

- **=**
  Equal to
- **>**
  Greater than
- **>=**
  Greater than or equal to
- **<**
  Less than
- **<=**
  Less than or equal to
- **!=**
  Not equal to

Value

Specifies the value of how to filter the field. You can either enter the value manually in the text box or select a value from the drop-down list. When you type in the value manually, if multiple values are required, they should be separated with ",", and if "," or "\" is contained in the values, write it as "\," or "\\".

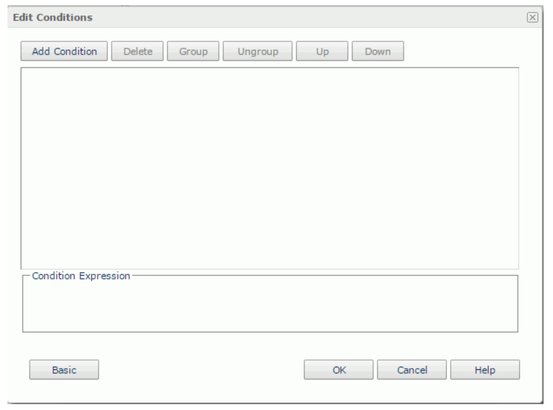
Logic

Lists the logic operator.

- **AND**
  Logic operator AND which is applied to this and the next line.
- **OR**
  Logic operator OR which is applied to this and the next line.
- **END**
  When this is the last line, use END to end the whole condition expression.

Advanced mode

The advanced mode enables the building of more complex filter conditions via the grouping of conditions.



Add Condition

Adds a new condition line. Each condition line contains an expression which is composed of a field, an operator, and a value.

Delete

Deletes the selected condition lines and groups.

Group

Makes the selected condition lines become a group. A group can have one logic operator to connect all of its condition lines, for example, a group contains three conditions lines which are expression A, B and C and the group's logic operator is OR, then the group's expression is: A OR B OR C.

Condition lines can also be added to an existing group by selecting the condition lines and the group while holding the Ctrl button, and then clicking the Group button.

Ungroup

Makes the selected condition lines and groups out of a group or disbands a group.

Up

Moves the selected condition line or group up to a higher level.

Down

Moves the selected condition line or group down to a lower level.

Logic

Specifies the logic operator of a condition group. By clicking the logic button the following items will be rolled one by one.

- **AND**
  Logic operator AND which is used to connect all of the condition lines in the group.
- **OR**
  Logic operator OR which is used to connect all of the condition lines in the group.
- **AND NOT**
  Logic operator AND NOT which is used to connect all of the condition lines in the group.
- **OR NOT**
  Logic operator OR NOT which is used to connect all of the condition lines in the group.

Field

Specifies the field to be filtered.

Operator

Specifies the operator to compose the filter expression.

Value

Specifies the value of how to filter the field. You can either enter the value manually in the text box or select a value from the drop-down list. When you type in the value manually, if multiple values are required, they should be separated with ",", and if "," or "\" is contained in the values, write it as "\," or "\\".

Condition Expression

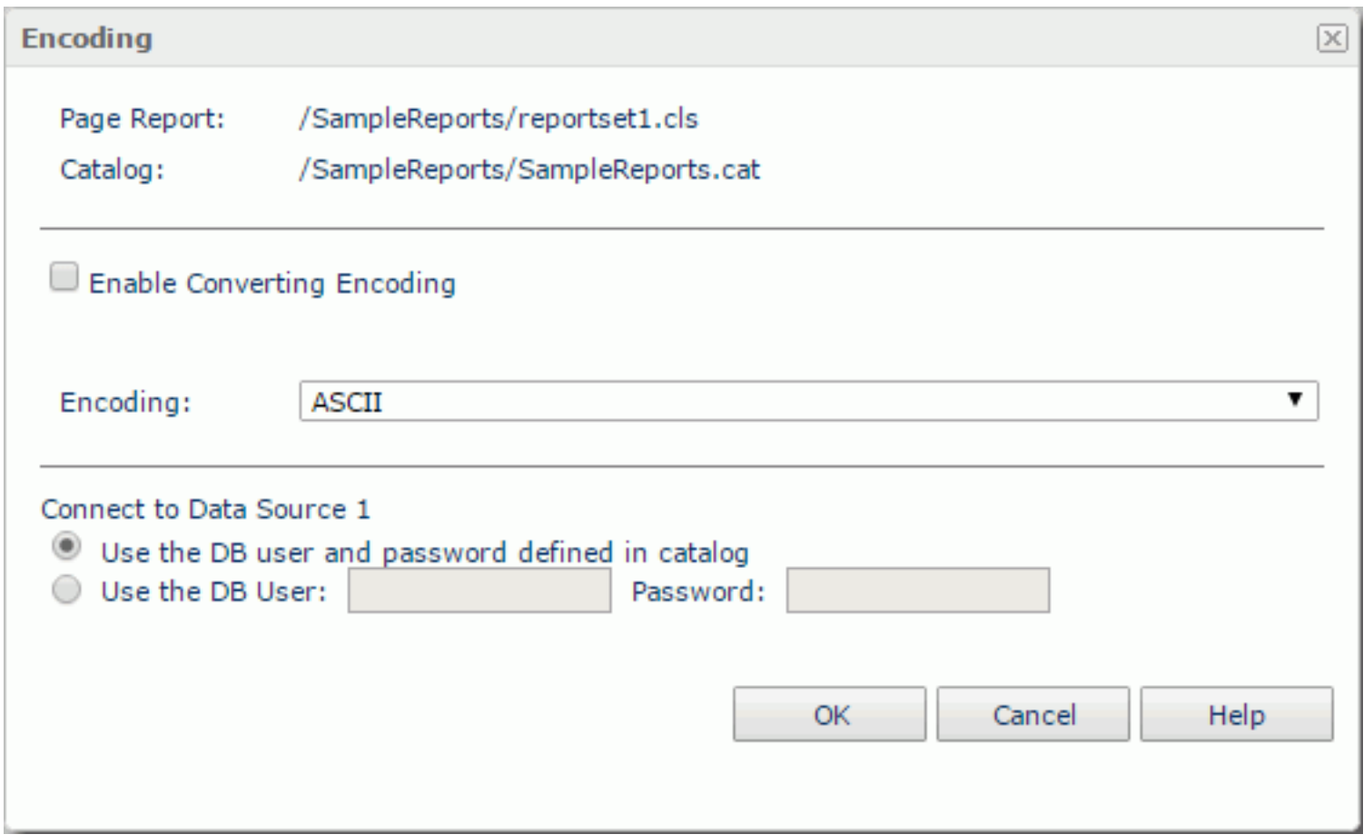
Displays the SQL statement of the filter condition.

Related topics:

- Adding conditional formats to fields

Encoding dialog

When a report designer links a trigger field of a master report to a detail report in JReport Designer, if he or she hasn't checked the option Use the same encoding and DB settings for the detail report as that of the mater report, then when the report is published to JReport Server and run in Page Report Studio, when you click the trigger field, this dialog appears for you to specify the encoding for the detail report.



Page Report

Displays the path and name of the current page report.

Catalog

Displays the path and name of the current catalog.

Enable Converting Encoding

Specifies to enable the conversion of encoding. From the Before Converting and After Converting drop-down lists, select the encoding as required.

Encoding

Specifies the encoding for the detail report.

Connect to [Data Source Name]

Specifies the DB user and password with which you want to connect to the data source used by the detail report.

- **Use the DB user and password defined in catalog**
  If checked, the DB user and password defined in the catalog will be used.
- **Use the DB User**
  If checked, specify another DB user and password instead of the one defined in the catalog.

OK

Closes this dialog and applies the settings to run the detail report.

Cancel

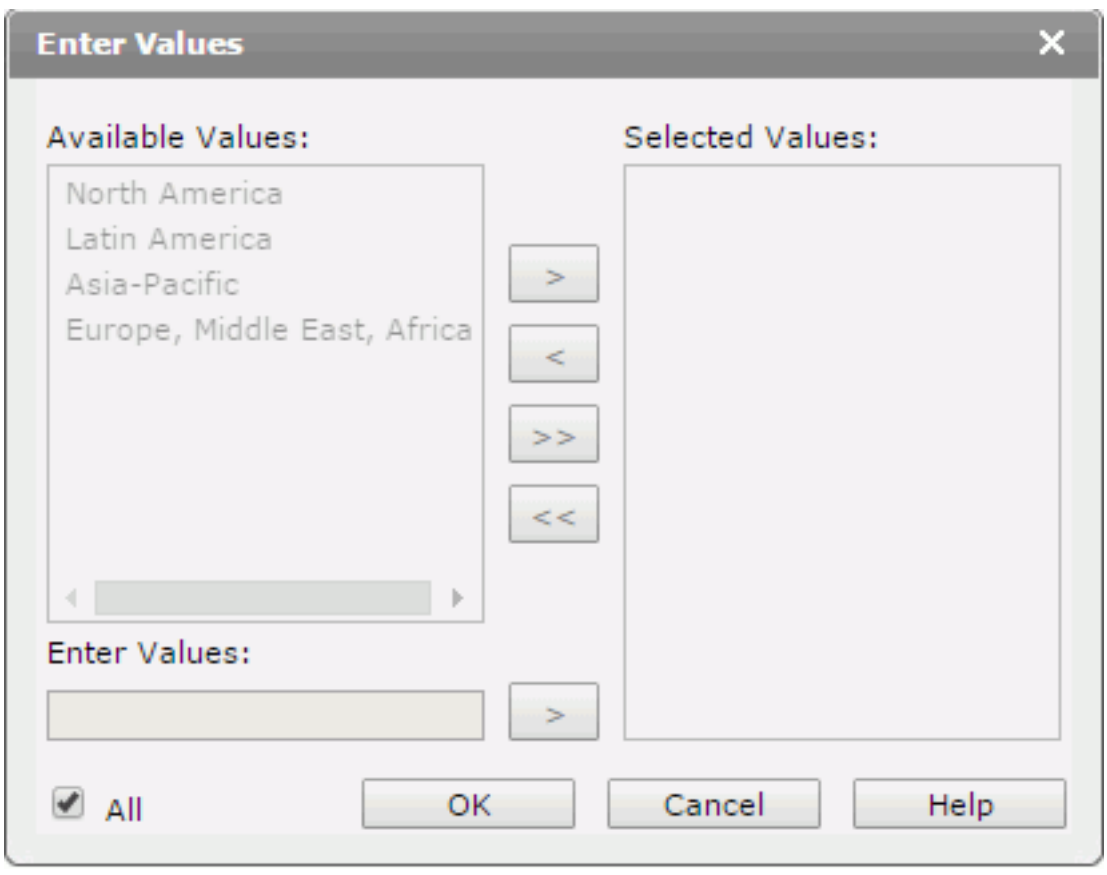
Closes this dialog and runs the detail report without regard to the settings.

Help

Displays the help document about this feature.

Enter Values dialog

The dialog helps you to specify multiple values for a parameter.



Available Values

Lists all predefined parameter values for selection. When the parameter is bound with a column, but the Display Column is different from the Bind Column, values of the Display Column are listed here.

Selected Values

Lists the values selected. The selected values are case sensitive.

Adds the selected values from the Available Values box to the Selected Values box.

Removes the selected values from the Selected Values box.

Adds all the selected values from the Available Values box to the Selected Values box.

Removes all the selected values from the Selected Values box.

Enter Values

This option is available when the parameter's Allow Type-in of Value property is set to true in JReport Designer.

Enter a value manually in the text box and then click the button next to add the value to the Selected Values box. When the parameter is bound with a column, but the display column is different from the bind column, make sure the value you enter is that of the bind column.

Opens the Calendar dialog to specify a date and time value. Available only for Date, DateTime, or Time parameter which allows for type-in values and is not bound with any column.

Search

This option is available when the parameter's Allow Type-in of Value property is set to false in JReport Designer. It is used to search for values among the available values. After you type text in the Search text box, the matched text in the available values will be highlighted.

All

If selected, it means that the parameter value result is all the values in the DBMS. This option is available when the parameter's Enable the "All" Option property is set to true in JReport Designer. This is translated in SQL to remove the parameter which may select more values than listed in the available values list.

For example, when the parameter query is:

```
SELECT CUSTOMERS.CUSTOMERID,CUSTOMERS.CUSTOMERNAME FROM CUSTOMERS where CUSTOMERS.
CUSTOMERID>0 and CUSTOMERS.CUSTOMERID<4
```

When you click

to add 1, 2 and 3 as the parameter values, which are all the available values, the SQL is:

```
(CUSTOMERS.CUSTOMERID IN ( 1,2,3))
```

Run the same report again and this time check the All checkbox in the Enter Values dialog, the query is then:

```
( 1 = 1)
```

In this case, you will get more customers even though available values are only 1 •3.

Note: When a multi-value parameter is inserted as a field into a report and All is selected as the value, the field will show the string "All". If the parameter allows type-in values, "All" means all possible values of the parameter data type; if the parameter does not, "All" represents all the values of the parameter that come from the DBMS.

OK

Applies the selected values to the parameter and closes the dialog.

Cancel

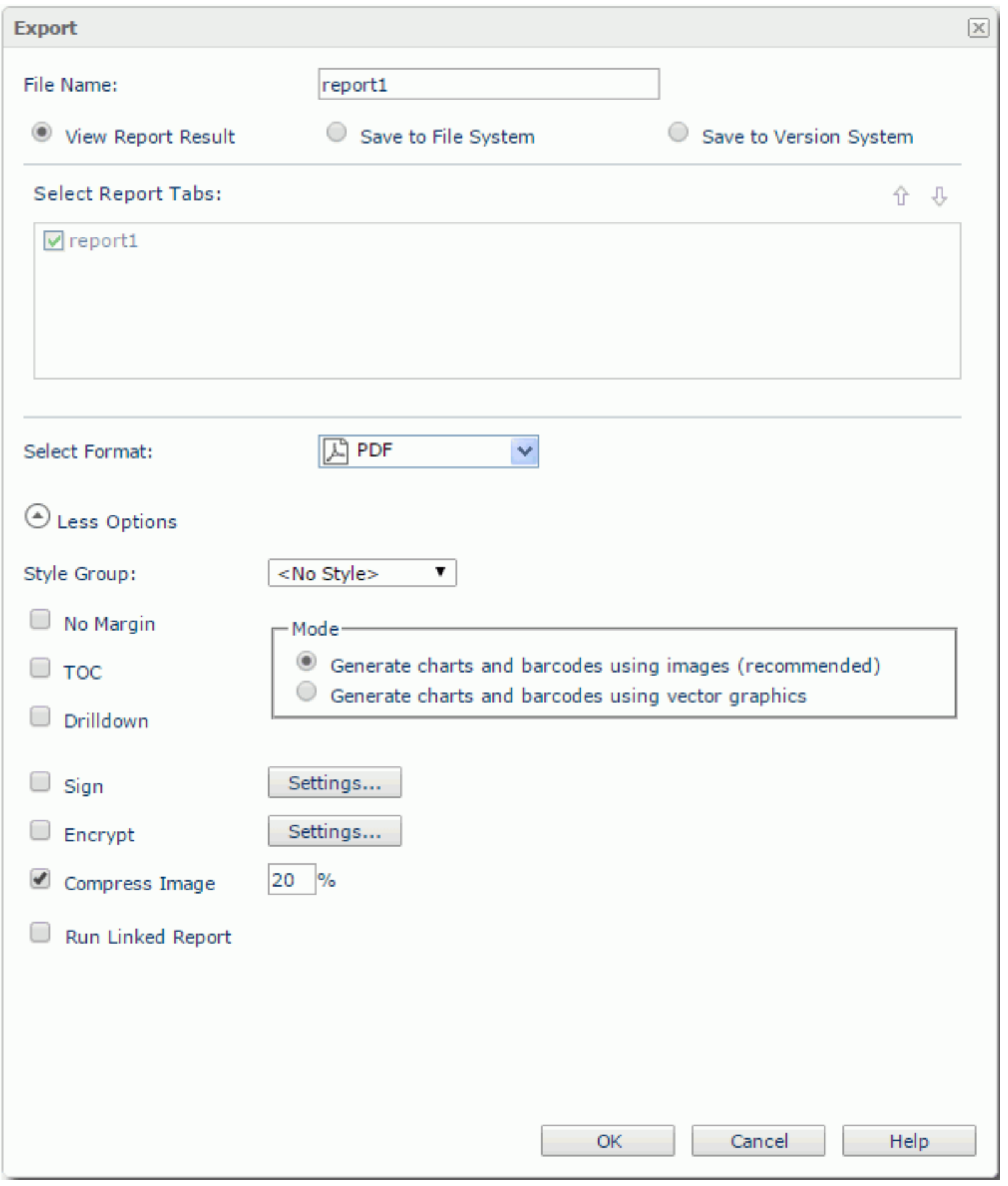
Does not retain any changes and closes the dialog.

Help

Displays the help document about this feature.

Export dialog

This dialog helps you to set settings for exporting the report result to different formats.



File Name

Specifies the name of the result file.

View Report Result

If selected, the result will be directly opened in the web browser if the format is supported by a plug-in of the web browser; otherwise you will be prompted to save the result file.

Save to File System

If selected, the web browser will prompt you to save the result file to a specified folder.

Save to Version System

If selected, the report result will be saved as a result version in JReport Server's versioning system.

Select Report Tabs

Specifies the report tabs which you want to export. The selected report tabs will be exported in the list order. If the report has only one report tab, it is selected by default.

Moves the specified report tab one step up.

Moves the specified report tab one step down.

Select Format

Specifies the format to which the report result will be exported.

More / Less Options

Shows/Hides the additional settings for exporting the report to the specified format.

- **Style Group**
  If a style group has been specified via the Override Style Group option in the Profile > Customize Server Preferences > Advanced tab, the specified style group will be selected in the drop-down list by default. You can also choose another style group from the drop-down list to apply to the exported result. If no style group is specified, the style group property predefined for specific export format in JReport Designer will be applied to export the report result to that format.

- **Properties**
  Specifies the properties for the selected format:
  - PDF
  - HTML
  - Excel
  - Text
  - RTF
  - XML
  - PostScript
  - **Page Report Result**
    - **Zip Result**
      Specifies to compress the result and its size would be smaller.

OK

Exports the report with the settings you specified. If you have selected multiple report tabs to export at one time and some of them haven't been specified with parameters, the Enter Parameter Values dialog will pop up for you to specify their parameter values.

Cancel

Cancels the operation and closes this dialog.

Help

Displays the help document about this feature.

Related topics:

- Exporting the result from a Page Report Studio window

Filter Control Properties dialog

This dialog helps you to edit the properties of a filter control. It contains the following tabs:

- General tab
- Font tab
- Border tab
- Title tab

OK

Applies the settings and closes this dialog.

Cancel

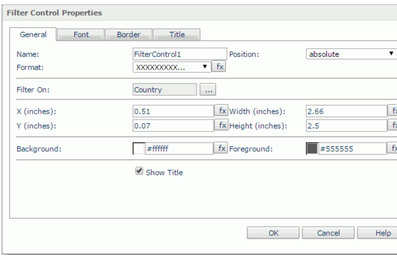
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the filter control.



Name

Specifies the name of the filter control.

Position

Displays the position mode of the filter control. If the filter control is directly contained in the report body, a tabular cell, or a text box, its position mode can be modified.

- **Absolute**: The filter control's position will be decided by its X and Y property values.
- **Static**: The filter control will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled.

Format

Specifies the field value format in the filter control.

Filter On

Displays the fields that the filter control is based on. You can click the button ... to open the Select Field dialog to select other fields.

X

Specifies the X coordinate of the filter control.

Y

Specifies the Y coordinate of the filter control.

Width

Specifies the width of the filter control.

Height

Specifies the height of the filter control.

Background

Specifies the background color of the filter control.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Foreground

Specifies the foreground color of the filter control.

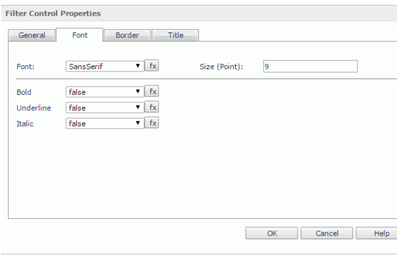
To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB.

Show Title

Specifies whether to show the title of the filter control.

Font tab

This tab shows the font-related information of the text in the filter control.



Font

Specifies the font face of the text.

Size

Specifies the font size of the text.

Bold

Specifies whether to make the text bold or not.

Underline

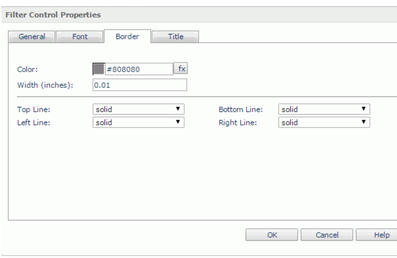
Specifies whether the text will be underlined or not.

Italic

Specifies whether to make the text italic or not.

Border tab

This tab shows information about borders of the filter control.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

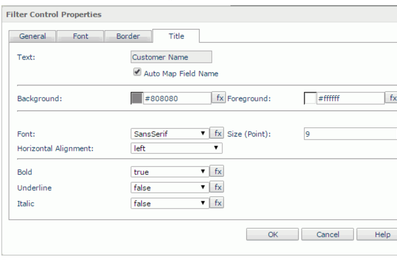
Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Title tab

This tab shows information about the title of the filter control. It is available when the Show Title option is selected in the General tab of the same dialog.



Text

Specifies the text of the title.

- **Auto Map Field Name**
  Specifies whether to use the field name as the title. If selected, the Text box above is disabled for inputting.

Background

Specifies the background color of the title.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Foreground

Specifies the foreground color of the title.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB.

Font

Specifies the font face of the text.

Size

Specifies the font size of the text.

Horizontal Alignment

Specifies the horizontal alignment mode of the text.

Bold

Specifies whether to make the text bold or not.

Underline

Specifies whether the text will be underlined or not.

Italic

Specifies whether to make the text italic or not.

Filter dialog

This dialog helps you to set criteria for filtering records in a report. It has the following two modes:

- Basic
- Advanced

Advanced/Basic

Switches the dialog to the advanced/basic mode.

OK

Accepts the changes, and regenerates the result using the filter settings with this dialog closed.

Cancel

Cancels the operation and closes this dialog.

Apply

Accepts the changes in the current tab, and regenerates the result using the filter settings.

Reset

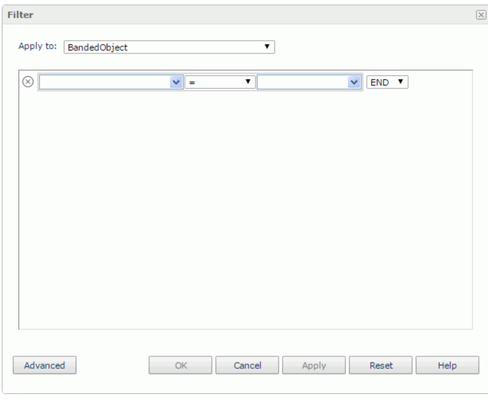
Restores this dialog to its opening status.

Help

Displays the help document about this feature.

Basic mode

The basic mode provides function for creating simple filter conditions which are connected by the AND and OR operators.



Apply to

Specifies the object to which the filter will be applied.

Deletes the current condition line.

Field

Specifies the field to be filtered.

Operator

Specifies the operator to compose the filter expression.

- **=**
  Equal to
- **>**
  Greater than
- **>=**
  Greater than or equal to
- **<**
  Less than
- **<=**
  Less than or equal to
- **!=**
  Not equal to
- **[not] in**
  Causes an enumerated list of values to appear in the WHERE clause predicate, used for evaluating for a true condition.
- **[not] like**
  Like string pattern matching operator is used to compare the first expression string value to the pattern string (the second expression). If you want to use wildcard character in the pattern string, only "_" and "%" are supported.
- **[not] between**
  Allows the system to evaluate whether or not data values are located between a range of values indicated in the predicate.
- **is [not] null**
  It is used in WHERE clause predicates to match null values occurring in a specified data field.

Value

Specifies the value of how to filter the field. You can either enter the value manually in the text box or select a value from the drop-down list. When you type in the value manually, if multiple values are required, they should be separated with ",", and if "," or "\" is contained in the values, write it as "\," or "\\".

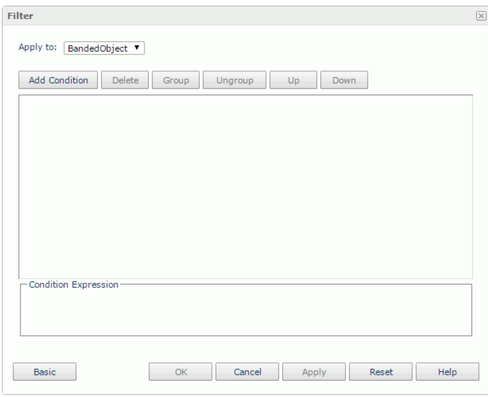
Logic

Lists the logic operator.

- **AND**
  Logic operator AND which is applied to this and the next line.
- **OR**
  Logic operator OR which is applied to this and the next line.

Advanced mode

The advanced mode enables the building of more complex filter conditions via the grouping of conditions.



Apply to

Specifies the object to which the filter will be applied.

Add Condition

Adds a new condition line.

Delete

Deletes the selected condition line.

Group

Makes the selected conditions in one group. Conditions can also be added to an existing group by selecting the conditions and the group while holding the Ctrl button, and then clicking the Group button.

Ungroup

Makes the selected condition ungrouped.

Up

Moves the selected condition or group up to a higher level.

Down

Moves the selected condition or group down to a lower level.

Logic

Specifies the logic operator.

- **AND**
  Logic operator AND which is applied to this and the next line.
- **OR**
  Logic operator OR which is applied to this and the next line
- **AND NOT**
  Logic operator AND NOT which is applied to this and the next line.
- **OR NOT**
  Logic operator OR NOT which is applied to this and the next line.

Field

Specifies the field to be filtered.

Operator

Specifies the operator to compose the filter expression.

Value

Specifies the value of how to filter the field. You can either enter the value manually in the text box or select a value from the drop-down list. When you type in the value manually, if multiple values are required, they should be separated with ",", and if "," or "\" is contained in the values, write it as "\," or "\\".

Condition Expression

Displays the SQL statement of the filter condition.

Related topics:

- Filtering the data components in a report

Format Category (X) Axis dialog

This dialog appears when you right-click a chart and then select Format Axis (unavailable to pie or indicator chart) > Format Category (X) Axis from the shortcut menu. It helps you to format the category (X) axis of the chart, and consists of the following tabs:

- Axis tab
- Tick Mark tab
- Font tab
- Orientation tab
- Format tab

OK

Applies the changes and closes the dialog.

Cancel

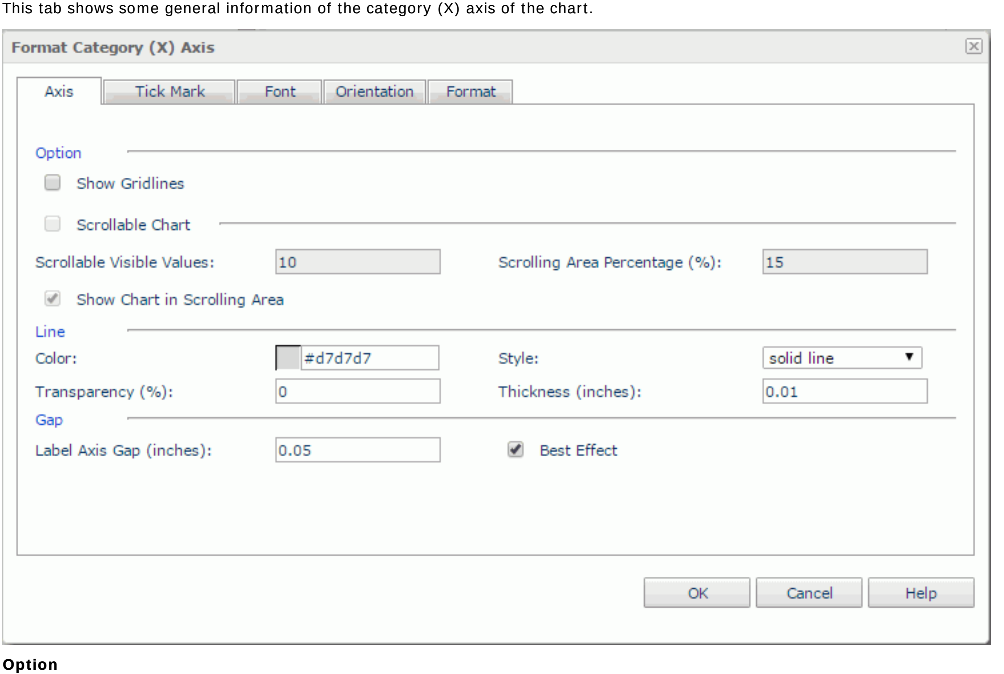
Does not retain any changes and closes the dialog.

Help

Displays the help document about this feature.

Axis tab

This tab shows some general information of the category (X) axis of the chart.



Option

Specifies the options for the axis.

- **Minimum Value**
  Specifies the minimum value that is to be displayed on the axis. Available only to bubble charts which use the axis to show numeric data and scatter charts.
- **Maximum Value**
  Specifies the maximum value that is to be displayed on the axis. Available only to bubble charts which use the axis to show numeric data and scatter charts.
- **Increment**
  Specifies the difference between two adjacent values on the axis. Available only to bubble charts which use the axis to show numeric data and scatter charts.
- **Number of Tick Marks**
  Specifies how many tick marks to be displayed on the axis. Available only to bubble charts which use the axis to show numeric data and scatter charts.
- **Show Gridlines**
  Specifies whether to show the horizontal gridlines in the chart.

Scrollable Chart

Specifies whether to make the chart scrollable. If checked, a scrollbar will be added in the chart, with which you can control the visible value range on the axis. Available to 2-D charts of bar, bench, line, area and stock types only.

- **Scrollable Visible Values**
  Specifies how many data items will be selected on the scrollbar and displayed on the axis by default.
- **Scrolling Area Percentage**
  Specifies the percentage the scrollbar occupies the whole size of the chart.
- **Show Chart in Scrolling Area**
  Specifies whether to show the thumbnail chart on the scrollbar.

Line

Specifies the line style for the axis.

- **Color**
  Specifies the color of the axis.
- **Style**
  Specifies the style for the line of the axis.
- **Transparency**
  Specifies the transparency for the color of the axis.
- **Thickness**
  Specifies the thickness for the line of the axis, in inches.

Gap

Specifies the gap properties for the labels on the axis.

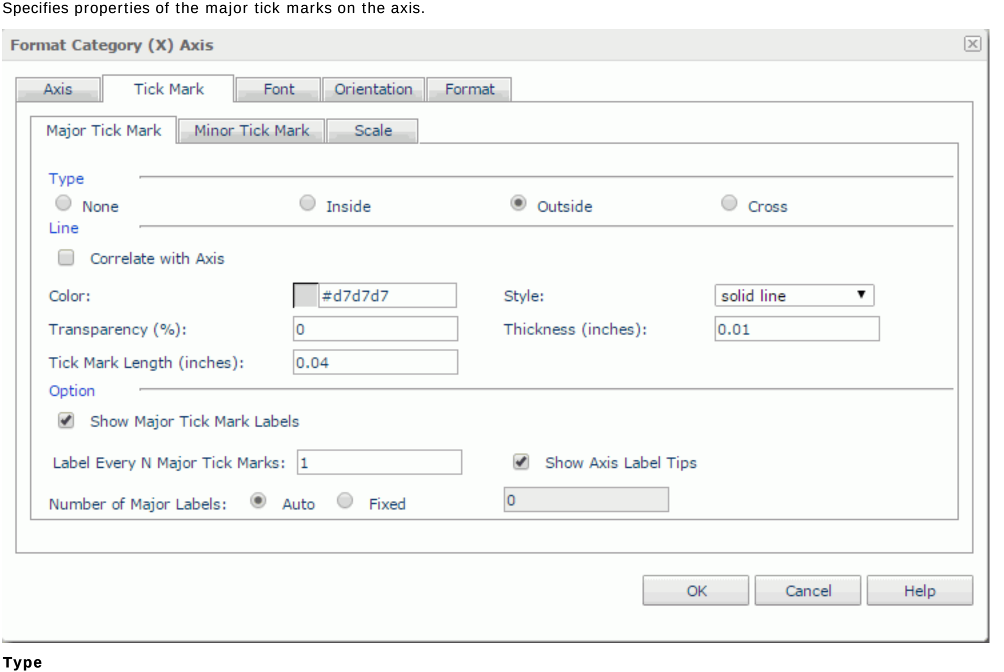
- **Label Axis Gap**
  Specifies the distance between the label and the axis, in inches.
- **Best Effect**
  Specifies whether to adjust the labels automatically to make them placed best.

Tick Mark tab

The tab consists of three sub tabs: major tick mark tab, minor tick mark tab, and scale tab.

Major Tick Mark tab

Specifies properties of the major tick marks on the axis.



Type

Specifies the type of the major tick marks on the axis.

- **None**
  If selected, major tick marks will not be shown on the axis and it will be meaningless to specify all the other major tick mark related properties.
- **Inside**
  If selected, major tick marks will be inside the chart.
- **Outside**
  If selected, major tick marks will be outside the chart.
- **Cross**
  If selected, major tick marks will be across the axis.

Line

Specifies the line properties of the major tick marks on the axis.

- **Correlate with Axis**
  If checked, the line properties of the major tick marks will correlate with that of the axis automatically.
  - **Color**
    Specifies the color of the major tick marks.
  - **Style**
    Specifies the type of the major tick marks.
  - **Transparency**
    Specifies the transparency for the color of the major tick marks.
  - **Thickness**
    Specifies the thickness of the major tick marks, in inches.
- **Tick Mark Length**
  Specifies the length of the major tick marks, in inches.

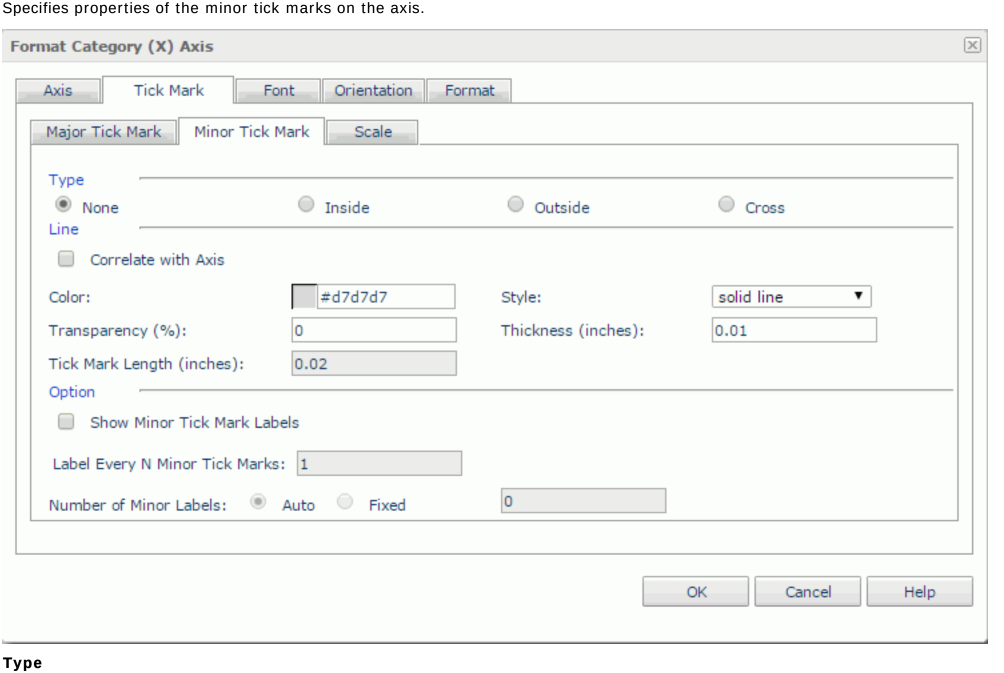
Option

Specifies the other properties of the major tick mark labels on the axis.

- **Show Major Tick Mark Labels**
  Specifies whether to display the labels of the major tick marks on the axis. If checked, the following properties will be enabled.
- **Label Every N Major Tick Marks**
  Specifies the frequency at which the major tick marks will be labeled.
- **Show Axis Label Tips**
  Specifies whether to display the complete label text when the mouse pointer points at a label on the axis.
- **Number of Major Labels**
  Specifies how many major tick mark labels to be displayed on the axis.
  - **Auto**
    If checked, all major tick mark labels will be shown.
  - **Fixed**
    If checked, you can specify the number of the major tick mark labels to be displayed on the axis.

Minor Tick Mark tab

Specifies properties of the minor tick marks on the axis.



Type

Specifies the type of the minor tick marks on the axis.

- **None**
  If selected, minor tick marks will not be shown on the axis and it will be meaningless to specify all the other minor tick mark related properties.
- **Inside**
  If selected, minor tick marks will be inside the chart.
- **Outside**
  If selected, minor tick marks will be outside the chart.
- **Cross**
  If selected, minor tick marks will be across the axis.

Line

Specifies the line properties for the minor tick marks on the axis.

- **Correlate with Axis**
  If checked, the line properties of the minor tick marks will correlate with that of the axis automatically.
  - **Color**
    Specifies the color of the minor tick marks.
  - **Style**
    Specifies the type of the minor tick marks.
  - **Transparency**
    Specifies the transparency of the minor tick marks.
  - **Thickness**
    Specifies the thickness of the minor tick marks, in inches.
- **Tick Mark Length**
  Specifies the length of the minor tick marks, in inches.

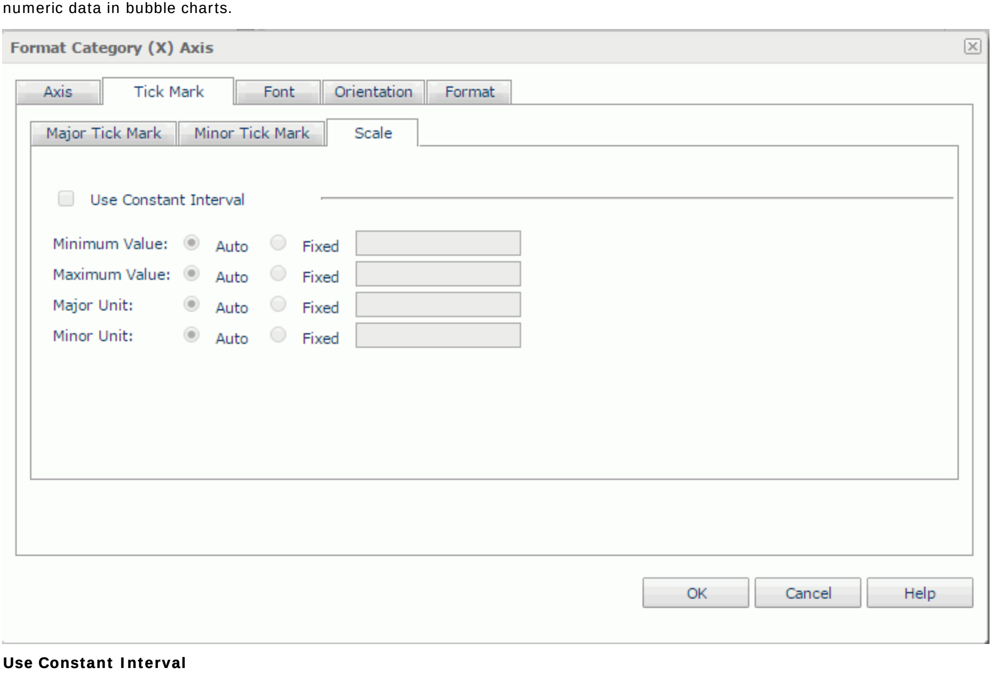
Option

Specifies the other properties of the minor tick marks on the axis.

- **Show Minor Tick Mark Labels**
  Specifies whether to display the labels of the minor tick marks on the axis. If checked, the following two properties will be enabled.
- **Label Every N Minor Tick Marks**
  Specifies the frequency at which the minor tick marks will be labeled.
- **Number of Minor Labels**
  Specifies how many minor tick mark labels to be displayed on the axis.
  - **Auto**
    If checked, all minor tick mark labels will be shown.
  - **Fixed**
    If checked, you can specify the number of the minor tick mark labels to be displayed on the axis.

Scale tab

Customizes the way in which to label the tick marks on the axis. Available only when the field on the category axis is one of the following types: Number, Date, DateTime, and Time, and not applied for scatter charts or when the category axis is used to show numeric data in bubble charts.



Use Constant Interval

Specifies whether to use a constant interval to label the tick marks. If checked, the values of the tick marks will be increased continually on the axis based on the following properties, instead of just using the data values.

Minimum

Specifies the minimum value which will be used to label the tick marks.

- **Auto**
  If checked, the minimum value will be defined by JReport automatically.
- **Fixed**
  If checked, you can define the minimum value as required. Input the value in the text box, or specify it in the calendar if the field on the category axis is of Date, DateTime or Time type.

Maximum

Specifies the maximum value which will be used to label the tick marks.

- **Auto**
  If checked, the maximum value will be defined by JReport automatically.
- **Fixed**
  If checked, you can define the maximum value as required. Input the value in the text box, or specify it in the calendar if the field on the category axis is of Date, DateTime or Time type.

Major Unit

Specifies the unit between two adjacent major tick marks.

- **Auto**
  If checked, the unit will be defined by JReport automatically.
- **Fixed**
  If checked, you can define the unit as required. Input the value in the text box, or choose the desired one from the drop-down list if the field on the category axis is of Date, DateTime or Time type.

Minor Unit

Specifies the unit between two adjacent minor tick marks.

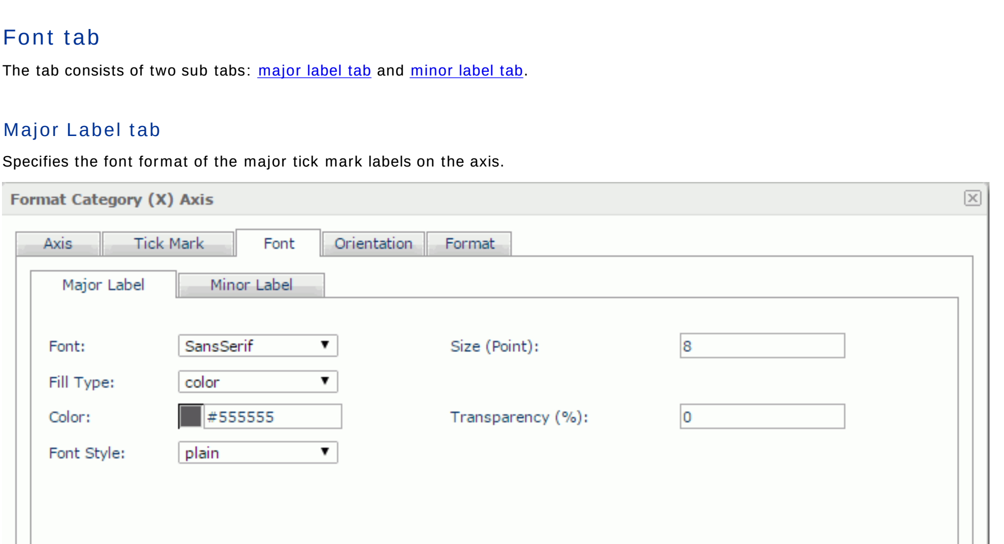
- **Auto**
  If checked, the unit will be defined by JReport automatically.
- **Fixed**
  If checked, you can define the unit as required. Input the value in the text box, or choose the desired one from the drop-down list if the field on the category axis is of Date, DateTime or Time type.

Font tab

The tab consists of two sub tabs: major label tab and minor label tab.

Major Label tab

Specifies the font format of the major tick mark labels on the axis.



Font

Specifies the font face of the label text.

Size

Specifies the font size of the label text.

Fill Type

Specifies the fill type of the label text.

Color

Specifies the color of the label text.

Transparency

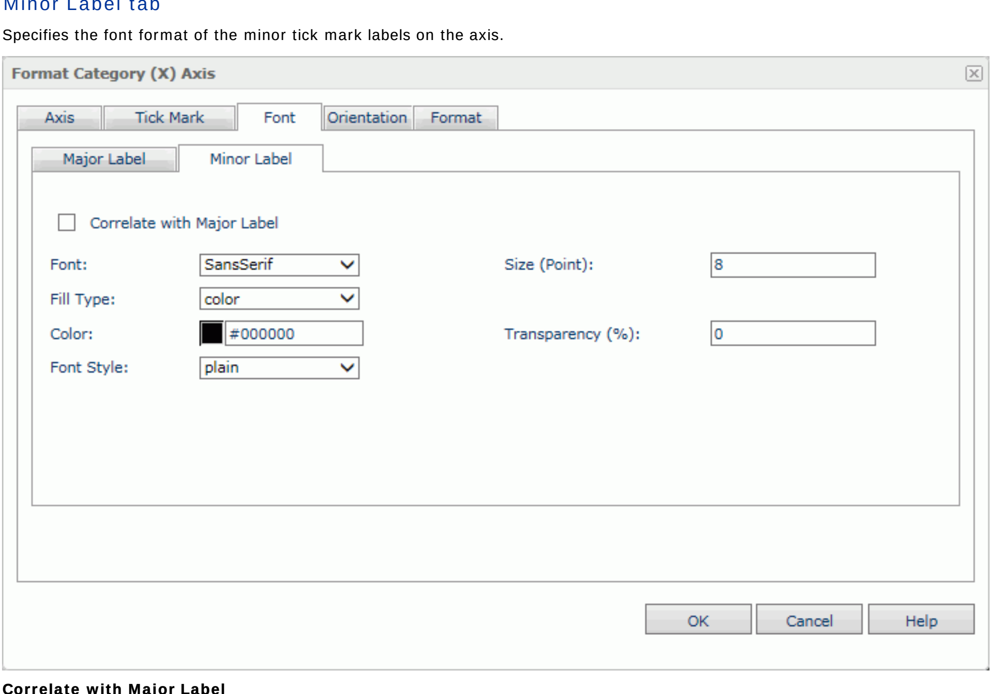
Specifies the transparency of the label text.

Font Style

Specifies the font style of the text. It can be one of the following: Plain, Bold, Italic and Bold Italic.

Minor Label tab

Specifies the font format of the minor tick mark labels on the axis.



Correlate with Major Label

If checked, the font format of the minor tick mark labels will correlate with that of the major tick mark labels automatically. Only when it is unchecked can the font properties of the minor tick mark labels take effect.

Font

Specifies the font face of the label text.

Size

Specifies the font size of the label text.

Fill Type

Specifies the fill type of the label text.

Color

Specifies the color of the label text.

Transparency

Specifies the transparency of the label text.

Font Style

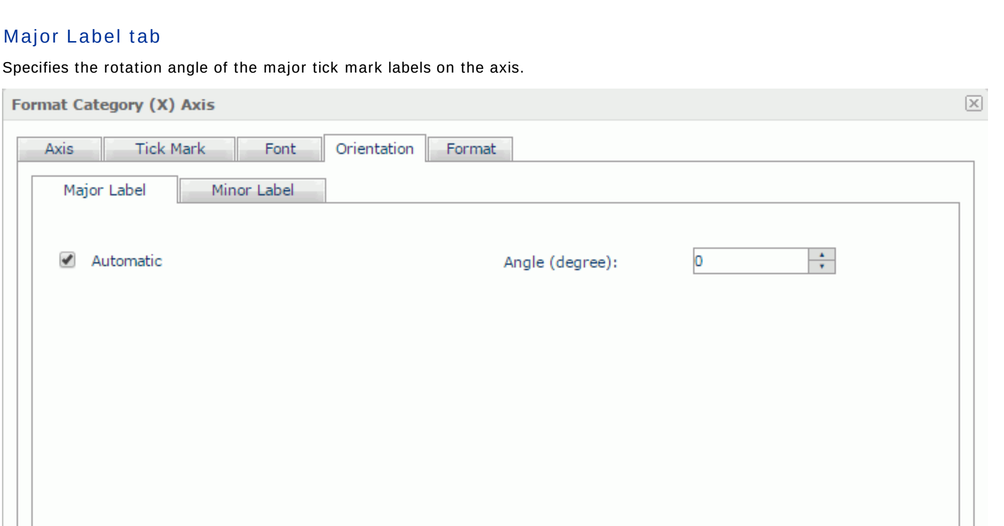
Specifies the font style of the text. It can be one of the following: Plain, Bold, Italic and Bold Italic.

Orientation tab

The tab consists of two sub tabs: major label tab and minor label tab.

Major Label tab

Specifies the rotation angle of the major tick mark labels on the axis.



Automatic

Specifies to adjust the rotation angle of the major tick mark label text on the axis automatically according to the length of the label text, in degrees.

When this option is checked by default:

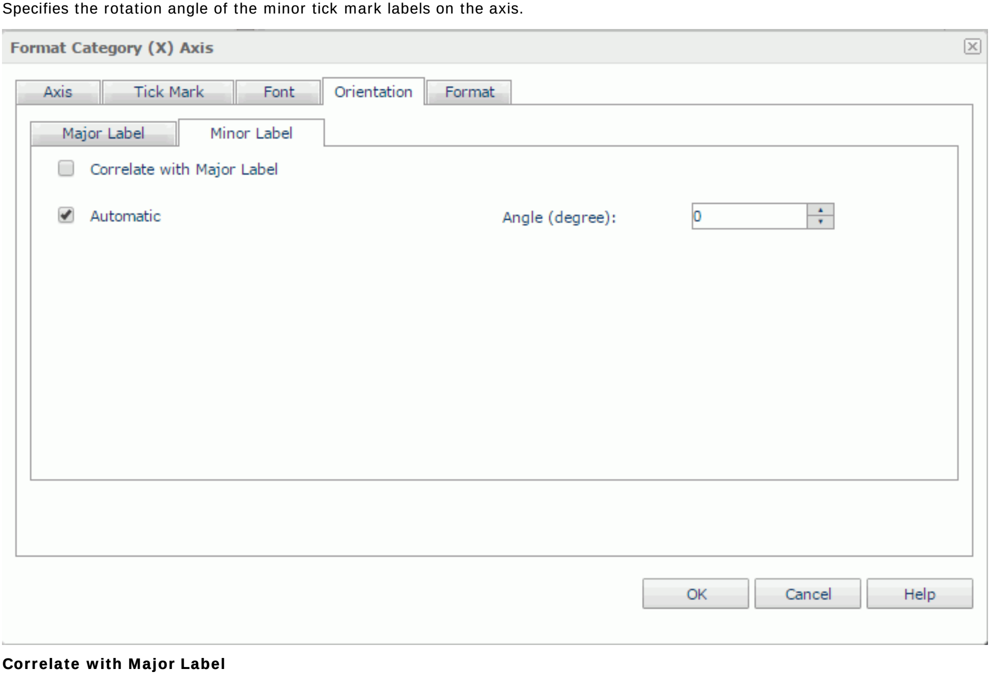
- If the text can be completely displayed horizontally, the default rotation angle will be 0.
- If the text cannot be completely displayed horizontally, the default rotation angle will be 30 anticlockwise, and the cut off part will be shown as suspension points.

Angle

Specifies to customize the rotation angle of the major tick mark label text on the axis.

Minor Label tab

Specifies the rotation angle of the minor tick mark labels on the axis.



Correlate with Major Label

If checked, the orientation setting of the minor tick mark labels will correlate with that of the major tick mark labels automatically. Only when it is unchecked can the orientation properties of the minor tick mark labels take effect.

Automatic

Specifies to adjust the rotation angle of the minor tick mark label text on the axis automatically according to the length of the label text, in degrees.

When this option is checked by default:

- If the text can be completely displayed horizontally, the default rotation angle will be 0.
- If the text cannot be completely displayed horizontally, the default rotation angle will be 30 anticlockwise, and the cut off part will be shown as suspension points.

Angle

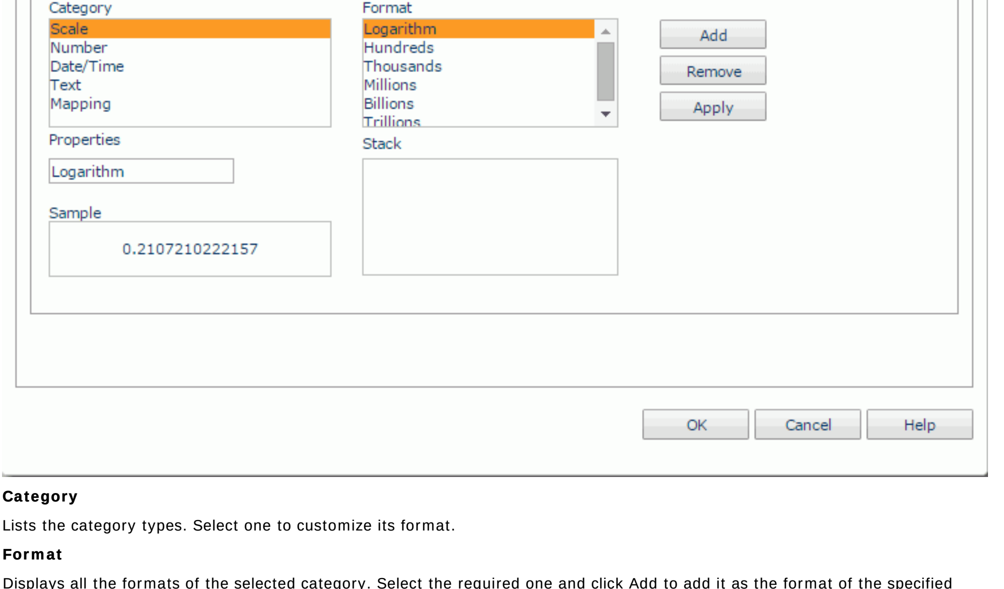
Specifies to customize the rotation angle of the minor tick mark label text on the axis.

Format tab

The tab consists of two sub tabs: major label tab and minor label tab.

Major Label tab

Specifies the data format of the major tick mark labels on the axis.



Category

Lists the category types. Select one to customize its format.

Format

Displays all the formats of the selected category. Select the required one and click Add to add it as the format of the specified category. You can add only one format for each category.

Properties

Displays the properties of the format you select. If the formats listed in the Format box cannot meet your requirement, define the format in the text field and then click Add to add it as the format of the specified category.

Sample

Displays the selected format effects.

Stack

Lists all the formats you select from different categories.

Add

Adds a format to the Stack list box.

Remove

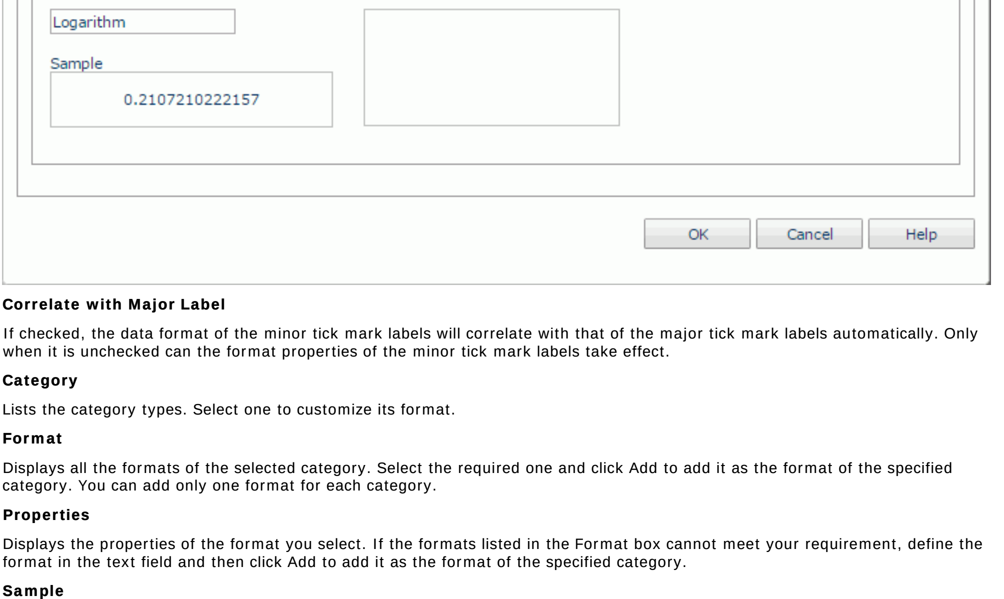
Removes a format from the Stack list box.

Apply

Applies the specified format in the Stack list box to the major tick mark labels.

Minor Label tab

Specifies the format of the minor tick mark labels on the axis.



Correlate with Major Label

If checked, the data format of the minor tick mark labels will correlate with that of the major tick mark labels automatically. Only when it is unchecked can the format properties of the minor tick mark labels take effect.

Category

Lists the category types. Select one to customize its format.

Format

Displays all the formats of the selected category. Select the required one and click Add to add it as the format of the specified category. You can add only one format for each category.

Properties

Displays the properties of the format you select. If the formats listed in the Format box cannot meet your requirement, define the format in the text field and then click Add to add it as the format of the specified category.

Sample

Displays the selected format effects.

Stack

Lists all the formats you select from different categories.

Add

Adds a format to the Stack list box.

Remove

Removes a format from the Stack list box.

Apply

Applies the specified format in the Stack list box to the minor tick mark labels.

Format Label dialog

This dialog helps you to format labels in a chart. It contains the following tabs:

- General tab
- Border tab
- Font tab

OK

Applies the settings and closes this dialog.

Cancel

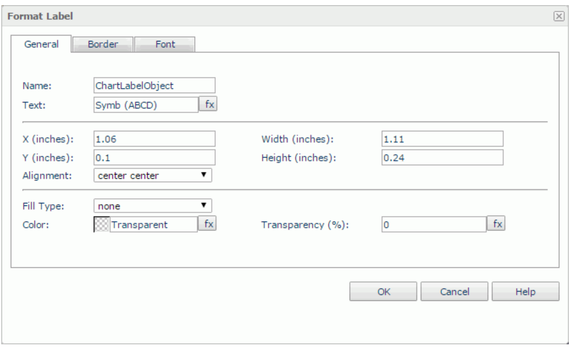
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the label.



Name

Specifies the display name of the label, which will be shown on the shortcut menu of the label.

Text

Specifies the label text.

X

Specifies the X coordinate of the label.

Y

Specifies the Y coordinate of the label.

Width

Specifies the width of the label.

Height

Specifies the height of the label.

Alignment

Specifies the alignment mode of the chart label.

Fill Type

Specifies the type for filling the label.

Color

Indicates the background color of the label.

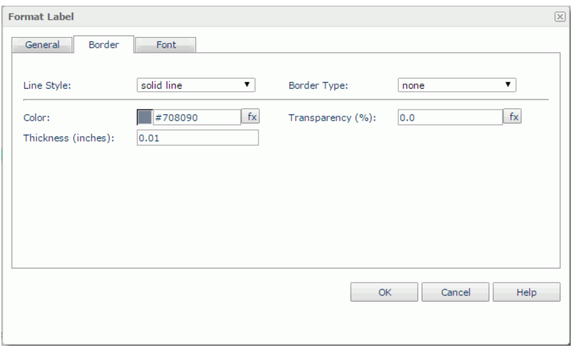
To change the color, click the color indicator to access the Select Color dialog, and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Transparency

Specifies the transparency of the label background.

Border tab

This tab shows information about borders of the label.



Line Style

Specifies the style of the label borders.

Border Type

Specifies the border type.

Color

Specifies the color of the label borders.

Transparency

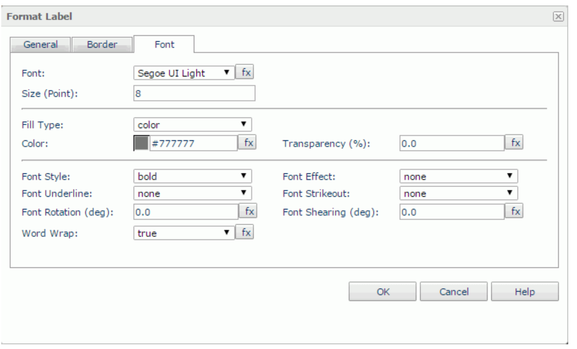
Specifies the transparency of the label borders.

Thickness

Specifies the thickness of the label borders.

Font tab

This tab shows the font-related information of the label.



Font

Specifies the font face of the label text.

Size

Specifies the font size of the label text.

Fill Type

Specifies the fill type of the label text.

Color

Specifies the color of the label text.

Transparency

Specifies the transparency of the label text.

Font Style

Specifies the font style of the label text.

Font Effect

Specifies the font effect of the label text.

Font Underline

Specifies the style of the line under the label text.

Font Strikeout

Specifies whether or not to attach a strikeout line to the label text.

Font Rotation

Specifies the rotation angle of the label around its center, in degrees.

Font Shearing

Specifies the shearing transformation of the label around its center, in degrees.

Word Wrap

Specifies whether or not to enable word wrapping for the label text.

Format Legend dialog

This dialog helps you to format the legend of a chart. It contains the following tabs:

- General tab
- Placement tab
- Border tab
- Font tab

OK

Applies the settings and closes this dialog.

Cancel

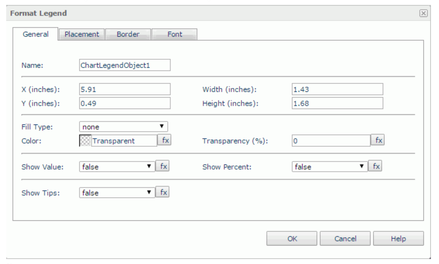
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the chart legend.



Name

Specifies the display name of the chart legend, which will be shown on the shortcut menu of the legend.

X

Specifies the X coordinate of the chart legend. Available only when the Placement option in the Placement tab is set to customized.

Y

Specifies the Y coordinate of the chart legend. Available only when the Placement option in the Placement tab is set to customized.

Width

Specifies the width of the chart legend. Available only when the Placement option in the Placement tab is set to customized.

Height

Specifies the height of the chart legend. Available only when the Placement option in the Placement tab is set to customized.

Fill Type

Specifies the type for filling the chart legend.

Color

Indicates the background color of the chart legend.

To change the color, click the color indicator to access the Select Color dialog, and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Transparency

Specifies the transparency of the legend background.

Show Value

Specifies whether to show the value of each legend.

Show Percent

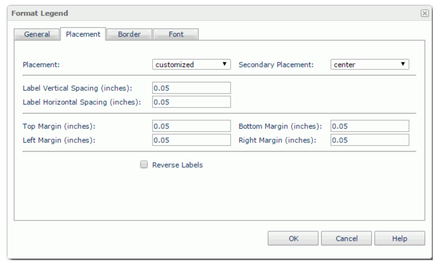
Specifies whether to show the percentage of each legend.

Show Tips

Specifies whether to show the corresponding data information when the mouse pointer points at a target in the chart legend.

Placement tab

This tab shows the position-related information of the chart legend.



Placement

Specifies the position of the legend to be left, right, top, bottom or customized by dragging on the chart manually.

Secondary Placement

Specifies the position of the legend on the basis of the Placement property.

Label Vertical Spacing

Specifies the vertical distance between two adjacent legend labels.

Label Horizontal Spacing

Specifies the horizontal distance between two adjacent legend labels.

Top Margin

Specifies the distance between the legend labels and the top border of the legend.

Bottom Margin

Specifies the distance between the legend labels and the bottom border of the legend.

Left Margin

Specifies the distance between the legend labels and the left border of the legend.

Right Margin

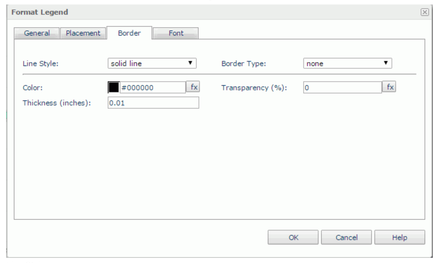
Specifies the distance between the legend labels and the right border of the legend.

Reverse Labels

Specifies whether or not to arrange the legend labels in a reverse order.

Border tab

This tab shows information about borders of the chart legend.



Line Style

Specifies the line style of the legend borders.

Border Type

Specifies the type of the legend borders.

Color

Specifies the color of the legend borders.

Transparency

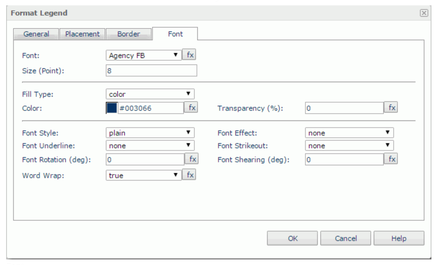
Specifies the transparency of the legend borders.

Thickness

Specifies the thickness of the legend borders.

Font tab

This tab shows the font-related information of the chart legend.



Font

Specifies the font face of the legend labels.

Size

Specifies the font size of the legend labels.

Fill Type

Specifies the fill type of the legend labels.

Color

Specifies the color of the legend labels.

Transparency

Specifies the transparency of the legend labels.

Font Style

Specifies the font style of the legend labels.

Font Effect

Specifies the font effect of the legend labels.

Font Underline

Specifies the style of the line under the legend labels.

Font Strikeout

Specifies whether or not to attach a strikeout line to the legend labels.

Font Rotation

Specifies the rotation angle of each legend label around its center, in degrees.

Font Shearing

Specifies the shearing transformation of each legend label around its center, in degrees.

Word Wrap

Specifies whether or not to enable word wrapping for the label text.

Format Paper dialog

This dialog helps you to format the paper of a chart. It contains the following tabs (some tabs may be not applicable for certain chart types):

- General tab
- Border tab
- Coordinates tab
- Graph tab
- Threshold Line tab
- Axis Z tab
- Bullet tab

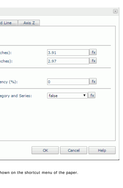
OK
Applies the settings and closes this dialog.

Cancel
Cancels the settings and closes this dialog.

Help
Displays the help document about this feature.

General tab

This tab shows some general information of the chart paper.



Name
Specifies the display name of the chart paper, which will be shown on the shortcut menu of the paper.

X
Specifies the X coordinate of the chart paper.

Y
Specifies the Y coordinate of the chart paper.

Width
Specifies the width of the chart paper.

Height
Specifies the height of the chart paper.

Fill Type
Specifies the type for filling the chart paper.

Color
Indicates the background color of the chart paper.

To change the color, click the color indicator to access the Select Color dialog, then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

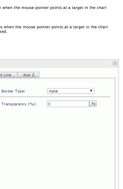
Transparency
Specifies the transparency of the chart paper background.

Show Tips
Specifies whether to show the corresponding data information when the mouse pointer points at a target in the chart paper.

Show Category and Series
Specifies whether to show data of the category and value axes when the mouse pointer points at a target in the chart paper. It takes effect only when the option Show Tips is checked.

Border tab

This tab shows information about borders of the chart paper.



Line Style
Specifies the line style of the chart paper borders.

Border Type
Specifies the type of the chart paper borders.

Color
Specifies the color of the chart paper borders.

Transparency
Specifies the transparency of the chart paper borders.

Thickness
Specifies the thickness of the chart paper borders.

Coordinate tab

You can use this tab to view and configure properties of the coordinates.

Scale X
Specifies the scaling ratio for the X axis.

Scale Y
Specifies the scaling ratio for the Y axis.

Scale Z
Specifies the scaling ratio for the Z axis.

Angle X
Specifies the rotation angle around the X axis. Applies to 3-D charts and pie charts.

Angle Y
Specifies the rotation angle around the Y axis. Applies to 3-D charts and pie charts.

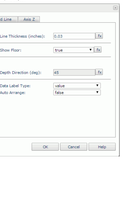
Angle Z
Specifies the rotation angle around the Z axis. Applies to 3-D charts and pie charts.

Perspective
Specifies the perspective effect of the chart. Input an integer to set the effect.

Interactive
Specifies whether the chart can be interactive or not.

Graph tab

You can use this tab to view and configure properties of the graphic object.



Bar Width
Specifies the width of the bars. Applies to bar charts only.

Line Thickness
Specifies thickness of the lines. Applies to line charts only.

Show Wall
Specifies whether or not to show the chart wall.

Show Floor
Specifies whether or not to show the chart floor.

Use Depth
Specifies whether to make the chart visually three-dimensional. Applies to 2-D bar, bench, line and area charts only.

- **Depth**
  Specifies the depth of the bars/benches/lines/areas.

- **Depth Direction**
  Specifies the angle of the axis along the depth of the bars/benches/lines/areas.

Show Static Data Label
Specifies whether or not to show the static data labels on the data markers. Applies to 2-D bar, bench, line, area, pie and ring charts only, and only when it is set to true can the following static data label related properties take effect.

Data Label Type
Specifies in which way the static data labels will be displayed.

- **Value** - Shows the value for each data marker.
- **Category Name** - Shows the category name for each pie section. Only applies to pie chart.
- **Percent** - Shows the percentage of each data marker to the total.
- **Value and Percent** - Shows the value and the percentage for each data marker.

Position
Specifies the position of the static data labels around the data markers.

- **Auto Arrange**
  If selected, the static data labels will be displayed automatically.
- **Outside Top**
  If selected, the static data labels will be displayed on the outside top of the bars. Available to bar and bench charts only.
- **Inside Top**
  If selected, the static data labels will be displayed on the inside top of the bars. Available to bar and bench charts only.
- **Inside Center**
  If selected, the static data labels will be displayed at the inside center of the bars. Available to bar and bench charts only.
- **Inside Bottom**
  If selected, the static data labels will be displayed at the inside bottom of the bars. Available to bar and bench charts only.
- **Top Center**
  If selected, the static data labels will be displayed on the top center of the line/area nodes. Available to line and area charts only.
- **Top Left**
  If selected, the static data labels will be displayed beside each section of the pies. Available to pie charts only.
- **Bottom Center**
  If selected, the static data labels will be displayed at the bottom center of the line/area nodes. Available to line and area charts only.
- **Bottom Right**
  If selected, the static data labels will be displayed at the bottom right of the line/area nodes. Available to line and area charts only.
- **Inside**
  If selected, the static data labels will be displayed beside each section of the pies. Available to pie charts only.
- **Slim Leg**
  If selected, the static data labels will be displayed beside each section and pointed by thin lines. Available to pie charts only.
- **Bestfit**
  If selected, the static data labels will be displayed at the best fit position automatically. Available to pie charts only.
- **Center**
  If selected, the static data labels will be displayed on each section (slice) of the pies. Available to pie charts only.

Auto Arrange
Specifies whether to display the static data labels inside the bars at the best position. Available only to bar and bench charts. If set to true the static data label's position is set to one of the following: Inside Center, Inside Top or Inside Bottom.

- **true**
  The static data label will be displayed horizontally at the specified position if the bar has enough room horizontally. Otherwise, it will be displayed vertically. If a bar does not have enough room both vertically and horizontally, its static data label will not be displayed.
- **false**
  The static data labels will be displayed at the specified position, and if the labels get overlapping, some of them will not be displayed.

Show Pie Name
Specifies whether or not to show the pie name. Applies to pie charts only.

Threshold Line tab

You can use this tab to view and configure properties of the threshold lines.

Show Threshold Line1
Specifies whether or not to show the first threshold line.

Threshold Value1
Specifies the value of the first threshold line.

Threshold Line Color1
Specifies the color of the first threshold line.

Show Threshold Line2
Specifies whether or not to show the second threshold line.

Threshold Value2
Specifies the value of the second threshold line.

Threshold Line Color2
Specifies the color of the second threshold line.

Transparency
Specifies the transparency of the threshold lines.

Axis Z tab

You can use this tab to view and configure properties of the Z axis.

Show Axis Z
Specifies whether or not to show the Z axis.

Placement
Specifies the position of the axis.

Show Axis Label Tips
Specifies whether to display the complete label text when the mouse pointer points at a label on the axis.

Label Font Automatic Orientation
Specifies to adjust the rotation angle of the label text on the axis automatically according to the length of the label text, in degree.

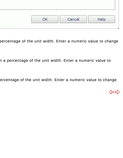
When this option is true by default:

- If the text can be completely displayed horizontally, the default rotation angle will be 0.
- If the text can not be completely displayed horizontally, the default rotation angle will be 30 anticlockwise, and the cut off part will be shown as suspension points.

Angle
Specifies to customize the rotation angle of the label text on the axis. Activated when Label Font Automatic Orientation is set to false (selected). You can input the angle in the text box directly.

Bullet tab

You can use this tab to view and configure properties of the bullets in a chart.



Featured Measure Width
Specifies the width of the featured measures, measured in a percentage of the unit width. Enter a numeric value to change the width.

Comparative Measure Width
Specifies the width of the comparative measures, measured in a percentage of the unit width. Enter a numeric value to change the width.

Qualitative Ranges Width
Specifies the width of the qualitative ranges, measured in a percentage of the unit width. Enter a numeric value to change the width.

Format Platform dialog

This dialog helps you to format the platform of a chart. It contains the following tabs:

- General tab
- Border tab
- Data tab
- Others tab

OK

Applies the settings and closes this dialog.

Cancel

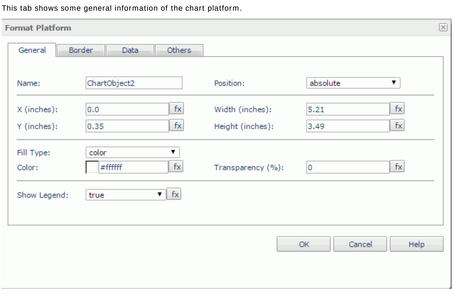
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the chart platform.



Name

Specifies the display name of the platform, which will be shown on the shortcut menu of the platform.

Position

Displays the position mode of the platform. If the platform is directly contained in the report body, a tabular cell, or a text box, its position mode can be modified.

- **Absolute**: The platform's position will be decided by its X and Y property values.
- **Static**: The platform will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled.

X

Specifies the X coordinate of the platform.

Y

Specifies the Y coordinate of the platform.

Width

Specifies the width of the platform.

Height

Specifies the height of the platform.

Fill Type

Specifies a type for filling the platform.

Color

Indicates the background color of the platform.

To change the color, click the color indicator to bring out the Select Color dialog, and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Transparency

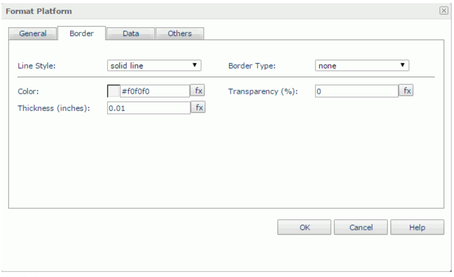
Specifies the transparency of the chart background.

Show Legend

Specifies whether or not to show the legend.

Border tab

This tab shows information about borders of the chart platform.



Line Style

Specifies the line style of the platform borders.

Border Type

Specifies the type of the platform borders.

Color

Specifies the color of the platform borders.

Transparency

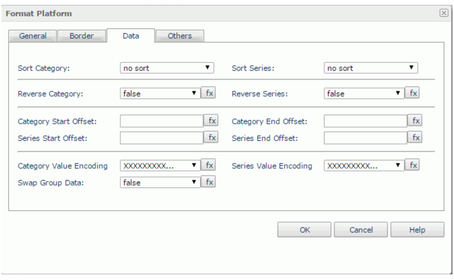
Specifies the transparency of the platform borders.

Thickness

Specifies the thickness of the platform borders.

Data tab

You can use this tab to view and configure properties of the chart data.



Sort Category

Specifies the sorting order for the category field values.

Sort Series

Specifies the sorting order for the series field values.

Reverse Category

Specifies whether or not to reverse the category field value sequence.

Reverse Series

Specifies whether or not to reverse the series field value sequence.

Category Start Offset

Specifies the starting offset of the categories.

Category End Offset

Specifies the ending offset of the categories.

Series Start Offset

Specifies the starting offset of the series.

Series End Offset

Specifies the ending offset of the series.

Category Value Encoding

Specifies the encoding format for values on the category axis. Formats here usually refer to the following: BIG5, EUCJIS, GBK, UTF8, and XXXXX....

Series Value Encoding

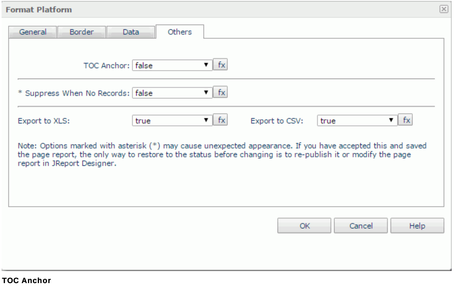
Specifies the encoding format for values on the series axis. Formats here usually refer to the following: BIG5, EUCJIS, GBK, UTF8, and XXXXX....

Swap Data Group

Specifies to display values from different data fields by switching data between the category and series axes, the category and values axes.

Others tab

You can use this tab to view and configure some miscellaneous settings.



TOC Anchor

Specifies whether or not to add the node that represents the platform to the TOC tree that is displayed in the TOC Browser.

Suppress When No Records

If true and no records are returned by the report, the platform will not be displayed.

Export to XLS

If true, the platform will be exported when you save the report result as an XLS file (make sure to check Data Format in the Export dialog).

Export to CSV

If true, the platform will be exported when you save the report result as a TXT file with Delimited Format selected.

Format Value (Y) Axis dialog

This dialog appears when you right-click a chart and select Format Axis (unavailable to pie or indicator chart) > Format Value (Y) Axis from the shortcut menu. It helps you to format the value (Y) axis of the chart and contains the following tabs:

- Axis tab
- Tick Mark tab
- Font tab
- Orientation tab
- Format tab

OK

Applies the changes and closes the dialog.

Cancel

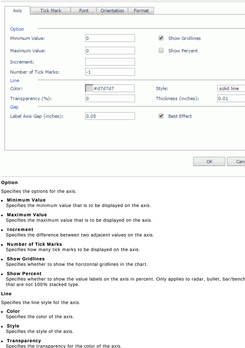
Does not retain any changes and closes the dialog.

Help

Displays the help document about this feature.

Axis tab

Specifies the general properties for the axis.



Option

Specifies the options for the axis.

- **Minimum Value**
  Specifies the minimum value that is to be displayed on the axis.
- **Maximum Value**
  Specifies the maximum value that is to be displayed on the axis.
- **Increment**
  Specifies the difference between two adjacent values on the axis.
- **Number of Tick Marks**
  Specifies how many tick marks to be displayed on the axis.
- **Show Gridlines**
  Specifies whether to show the horizontal gridlines in the chart.
- **Show Percent**
  Specifies whether to show the value labels on the axis in percent. Only applies to radar, bullet, bar/bench, line, and area chart that are not 100% stacked type.

Line

Specifies the line style for the axis.

- **Color**
  Specifies the color of the axis.
- **Style**
  Specifies the style of the axis.
- **Transparency**
  Specifies the transparency for the color of the axis.
- **Thickness**
  Specifies the thickness of the axis, in pixels.

Gap

Specifies the gap properties for the labels on the axis.

- **Label Axis Gap**
  Specifies the distance between the label and the axis, in pixels.
- **Best Effect**
  Specifies whether to adjust the labels on the axis automatically to make them placed best.

Tick Mark tab

The tab consists of two sub tabs: major tick mark tab, and minor tick mark tab.

Major Tick Mark tab

Specifies properties of the major tick marks on the axis.

Type

Specifies the type of the major tick marks on the axis.

- **None**
  If selected, major tick marks will not be shown on the axis and it will be meaningless to specify all the other major tick mark related properties.
- **Inside**
  If selected, major tick marks will be inside the chart.
- **Outside**
  If selected, major tick marks will be outside the chart.
- **Cross**
  If selected, major tick marks will be across the axis.

Line

Specifies the line properties of the major tick marks on the axis.

- **Correlate with Axis**
  If checked, the line properties of the major tick marks will correlate with that of the axis automatically.
  - **Color**
    Specifies the color of the major tick marks.
  - **Transparency**
    Specifies the transparency for the color of the major tick marks.
  - **Style**
    Specifies the type of the major tick marks.
  - **Thickness**
    Specifies the thickness of the major tick marks, in pixels.
- **Tick Mark Length**
  Specifies the length for the major tick marks, in pixels.

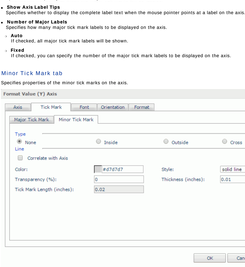
Option

Specifies the other properties of the major tick mark labels on the axis.

- **Show Major Tick Mark Labels**
  Specifies whether to display the labels of the major tick marks on the axis. If checked, the following properties will be enabled.
- **Label Every N Major Tick Marks**
  Specifies the frequency at which the major tick marks will be labeled.
- **Show Axis Label Tips**
  Specifies whether to display the complete label text when the mouse pointer points at a label on the axis.
- **Number of Major Labels**
  Specifies how many major tick mark labels to be displayed on the axis.
  - **Auto**
    If checked, all major tick mark labels will be shown.
  - **Fixed**
    If checked, you can specify the number of the major tick mark labels to be displayed on the axis.

Minor Tick Mark tab

Specifies properties of the minor tick marks on the axis.



Type

Specifies the type of the minor tick marks on the axis.

- **None**
  If selected, minor tick marks will not be shown on the axis and it will be meaningless to specify all the other minor tick mark related properties.
- **Inside**
  If selected, minor tick marks will be inside the chart.
- **Outside**
  If selected, minor tick marks will be outside the chart.
- **Cross**
  If selected, minor tick marks will be across the axis.

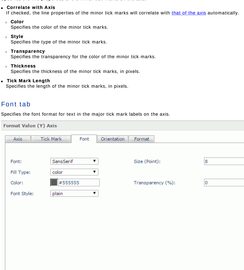
Line

Specifies the line properties of the minor tick marks on the axis.

- **Correlate with Axis**
  If checked, the line properties of the minor tick marks will correlate with that of the axis automatically.
  - **Color**
    Specifies the color of the minor tick marks.
  - **Style**
    Specifies the type of the minor tick marks.
  - **Transparency**
    Specifies the transparency for the color of the minor tick marks.
  - **Thickness**
    Specifies the thickness of the minor tick marks, in pixels.
- **Tick Mark Length**
  Specifies the length of the minor tick marks, in pixels.

Font tab

Specifies the font format for text in the major tick mark labels on the axis.



Font

Specifies the font face of the label text.

Size

Specifies the font size of the label text.

Fill Type

Specifies the fill type of the label text.

Color

Specifies the color of the label text.

Transparency

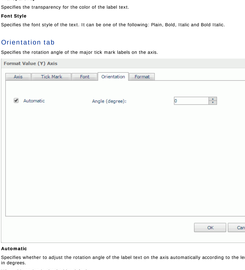
Specifies the transparency for the color of the label text.

Font Style

Specifies the font style of the text. It can be one of the following: Plain, Bold, Italic and Bold Italic.

Orientation tab

Specifies the rotation angle of the major tick mark labels on the axis.



Automatic

Specifies whether to adjust the rotation angle of the label text on the axis automatically according to the length of the label text, in degrees.

When this option is checked by default:

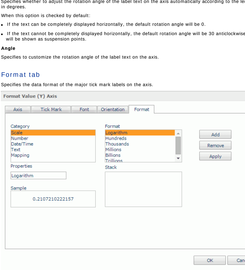
- if the text can be completely displayed horizontally, the default rotation angle will be 0.
- if the text cannot be completely displayed horizontally, the default rotation angle will be 30 anticlockwise, and the cut off part will be shown as suspension points.

Angle

Specifies to customize the rotation angle of the label text on the axis.

Format tab

Specifies the data format of the major tick mark labels on the axis.



Category

Lists the category types. Select one to customize its format.

Format

Displays all the formats of the selected category. Select the required one and click Add to add it as the format of the specified category. You can add only one format for each category.

Properties

Displays the properties of the format you select. If the formats listed in the Format box cannot meet your requirement, define the format in the text field and then click Add to add it as the format of the specified category.

Sample

Displays the selected format effects.

Stack

Lists all the formats you select from different categories.

Add

Adds a format to the Stack list box.

Remove

Removes a format from the Stack list box.

Apply

Applies the specified format in the Stack list box to the major tick mark labels.

Format Value (Y2) Axis dialog

This dialog appears when you right-click a chart and select Format Axis (unavailable to pie or indicator chart) > Format Value (Y2) Axis from the shortcut menu. It helps you to format the value (Y2) axis of the chart and contains the following tabs:

- Axis tab
- Tick Mark tab
- Font tab
- Orientation tab
- Format tab

OK

Applies the changes and closes the dialog.

Cancel

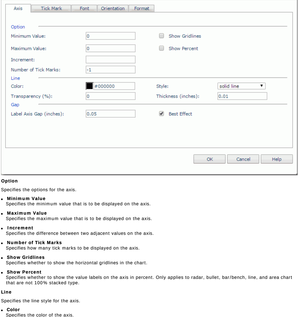
Does not retain any changes and closes the dialog.

Help

Displays the help document about this feature.

Axis tab

Specifies the general properties for the axis.



Option

Specifies the options for the axis.

- **Minimum Value**
  Specifies the minimum value that is to be displayed on the axis.
- **Maximum Value**
  Specifies the maximum value that is to be displayed on the axis.
- **Increment**
  Specifies the difference between two adjacent values on the axis.
- **Number of Tick Marks**
  Specifies how many tick marks to be displayed on the axis.
- **Show Gridlines**
  Specifies whether to show the horizontal gridlines in the chart.
- **Show Percent**
  Specifies whether to show the value labels on the axis in percent. Only applies to radar, bullet, barbench, line, and area chart that are not 100% stacked type.

Line

Specifies the line style for the axis.

- **Color**
  Specifies the color of the axis.
- **Style**
  Specifies the style of the axis.
- **Transparency**
  Specifies the transparency for the color of the axis.
- **Thickness**
  Specifies the thickness of the axis, in pixels.

Gap

Specifies the gap properties for the labels on the axis.

- **Label Axis Gap**
  Specifies the distance between the label and the axis, in pixels.
- **Best Effect**
  Specifies whether to adjust the labels on the axis automatically to make them placed best.

Tick Mark tab

The tab consists of two sub tabs: major tick mark tab, and minor tick mark tab.

Major Tick Mark tab

Specifies the properties of the major tick marks on the axis.

Type

Specifies the type of the major tick marks on the axis.

- **None**
  If selected, major tick marks will not be shown on the axis and it will be meaningless to specify all the other major tick mark related properties.
- **Inside**
  If selected, major tick marks will be inside the chart.
- **Outside**
  If selected, major tick marks will be outside the chart.
- **Cross**
  If selected, major tick marks will be across the axis.

Line

Specifies the line properties of the major tick marks on the axis.

- **Correlate with Axis**
  If checked, the line properties of the major tick marks will correlate with that of the axis automatically.
  - **Color**
    Specifies the color of the major tick marks.
  - **Transparency**
    Specifies the transparency for the color of the major tick marks.
  - **Style**
    Specifies the type of the major tick marks.
  - **Thickness**
    Specifies the thickness of the major tick marks, in pixels.
- **Tick Mark Length**
  Specifies the length for the major tick marks, in pixels.

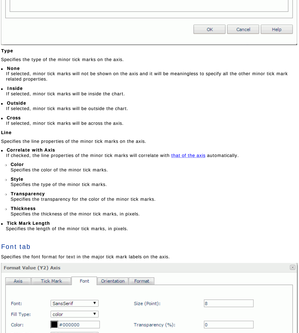
Option

Specifies the other properties of the major tick mark labels on the axis.

- **Show Major Tick Mark Labels**
  Specifies whether to display the labels of the major tick marks on the axis. If checked, the following properties will be enabled.
- **Label Every N Major Tick Marks**
  Specifies the frequency at which the major tick marks will be labeled.
- **Show Axis Label Tips**
  Specifies whether to display the complete label text when the mouse pointer points at a label on the axis.
- **Number of Major Labels**
  Specifies how many major tick mark labels to be displayed on the axis.
  - **Auto**
    If checked, all major tick mark labels will be shown.
  - **Fixed**
    If checked, you can specify the number of the major tick mark labels to be displayed on the axis.

Minor Tick Mark tab

Specifies the properties of the minor tick marks on the axis.



Type

Specifies the type of the minor tick marks on the axis.

- **None**
  If selected, minor tick marks will not be shown on the axis and it will be meaningless to specify all the other minor tick mark related properties.
- **Inside**
  If selected, minor tick marks will be inside the chart.
- **Outside**
  If selected, minor tick marks will be outside the chart.
- **Cross**
  If selected, minor tick marks will be across the axis.

Line

Specifies the line properties of the minor tick marks on the axis.

- **Correlate with Axis**
  If checked, the line properties of the minor tick marks will correlate with that of the axis automatically.
  - **Color**
    Specifies the color of the minor tick marks.
  - **Style**
    Specifies the type of the minor tick marks.
  - **Transparency**
    Specifies the transparency for the color of the minor tick marks.
  - **Thickness**
    Specifies the thickness of the minor tick marks, in pixels.
- **Tick Mark Length**
  Specifies the length of the minor tick marks, in pixels.

Font tab

Specifies the font format for text in the major tick mark labels on the axis.

Font

Specifies the font face of the label text.

Size

Specifies the font size of the label text.

Fill Type

Specifies the fill type of the label text.

Color

Specifies the color of the label text.

Transparency

Specifies the transparency for the color of the label text.

Font Style

Specifies the font style of the text. It can be one of the following: Plain, Bold, Italic and Bold Italic.

Orientation tab

Specifies the rotation angle of the major tick mark labels on the axis.

Automatic

Specifies whether to adjust the rotation angle of the label text on the axis automatically according to the length of the label text, in degrees.

When this option is checked by default:

- If the text can be completely displayed horizontally, the default rotation angle will be 0.
- If the text cannot be completely displayed horizontally, the default rotation angle will be 30 anticlockwise, and the cut off part will be shown as suspension points.

Angle

Specifies to customize the rotation angle of the label text on the axis.

Format tab

Specifies data format of the major tick mark labels on the axis.

Category

Lists the category types. Select one to customize its format.

Format

Displays all the formats of the selected category. Select the required one and click Add to add it as the format of the specified category. You can add only one format for each category.

Properties

Displays the properties of the format you select. If the formats listed in the Format box cannot meet your requirement, define the format in the text field and then click Add to add it as the format of the specified category.

Sample

Displays the selected format effects.

Stack

Lists all the formats you select from different categories.

Add

Adds a format to the Stack list box.

Remove

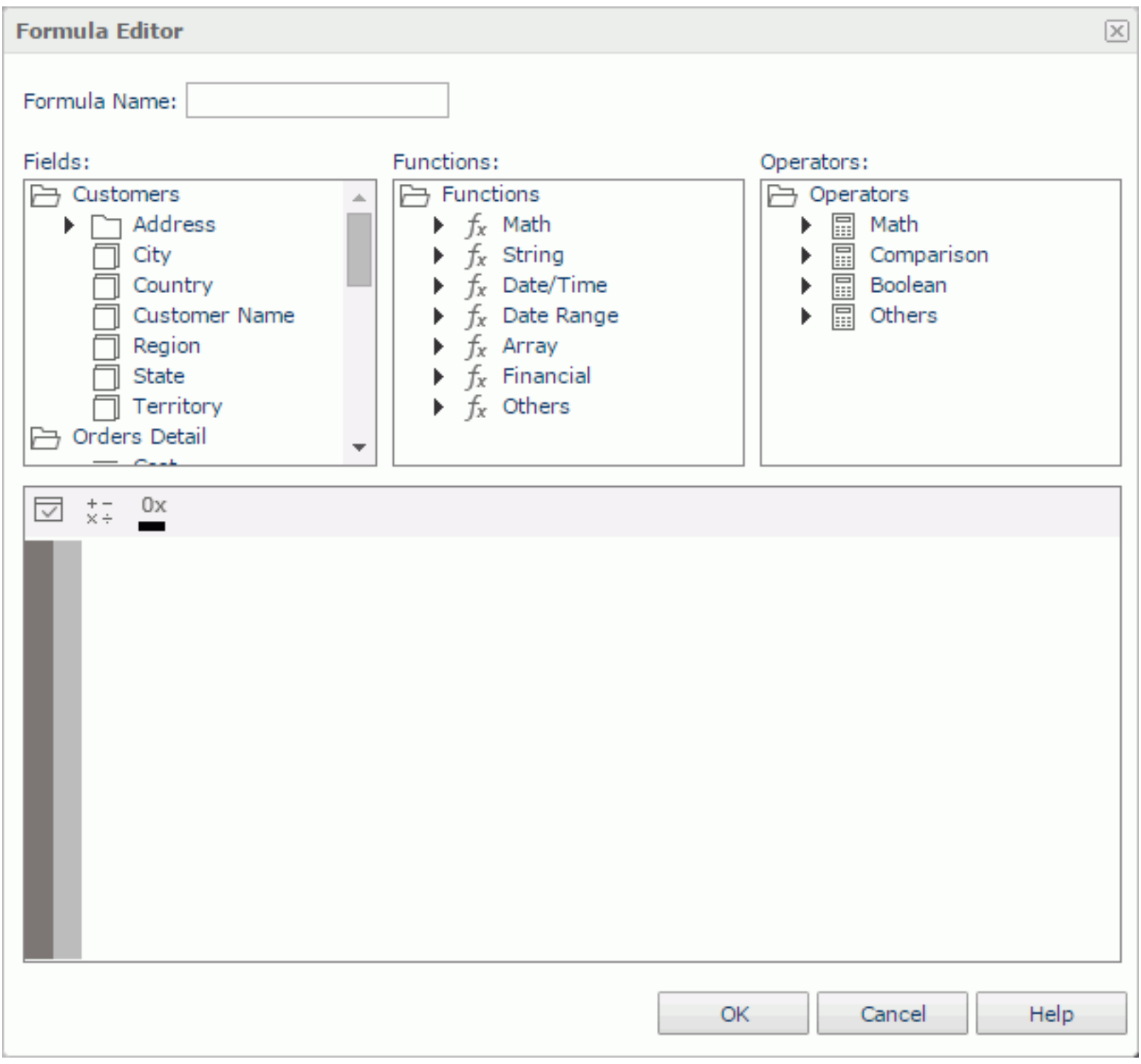
Removes a format from the Stack list box.

Apply

Applies the specified format in the Stack list box to the major tick mark labels.

Formula Editor dialog

This dialog helps you to create a dynamic formula in a report.



Formula Name

Specifies the name of the formula.

Fields box

Displays a list of those fields that are available for formulas. The fields include the group objects and detail objects in the current business view, and the dynamic formulas and aggregations that have been created in the report. You can select one field and double-click it to insert the field into the editing box at the insertion point.

Functions box

Displays a list of JReport functions that are available for formulas. When you select one function and double-click it, JReport will insert the selected function into the editing box at the insertion point completely with its required syntax items (parentheses, commas, and so on).

For details about usage of the functions, refer to Built-in functions.

Operators box

Displays a list of operators that are available for formulas. Select one operator and double-click it to insert the selected operator into the formula text box at the insertion point.

For details about usage of the operators, refer to Operators.

Editing box

In this box, you can build and edit your formula. There are several ways to work with formulas:

- Select formula components from the Fields, Functions and Operators boxes in the Formula Editor, and then double-click the components, JReport will then insert them in the formula;
- Type your formula in the editing box directly;
- Use the above two methods together;
- Paste formula text from the text document of other programs.

Note: If you refer to any field in the formula, the reference name for that field will be prefixed with an @ sign. If the field name contains spaces, the reference name in formula will be quoted with double-quotation marks (""). For example, if the field name is Customer Name, then the reference name will be @"Customer Name".

Check

Tests the syntax of your formula. If the syntax is incorrect, JReport provides an opportunity to correct the errors.

Add Operators

Selects a general operator to be used in the editing box.

Color Converter

A color palette is provided for you to insert the HEX code of a color simply by clicking the corresponding color in the color palette instead of inputting the HEX code manually.

OK

Creates the formula and adds it to the Resource View panel.

Cancel

Doesn't add the formula and closes the dialog.

Help

Displays the help document about this feature.

Related topics:

- Creating and using dynamic formulas

Group Properties dialog

This dialog helps you to edit the properties of a group object. It contains the following tabs:

- General tab
- Others tab

OK

Applies the settings and closes this dialog.

Cancel

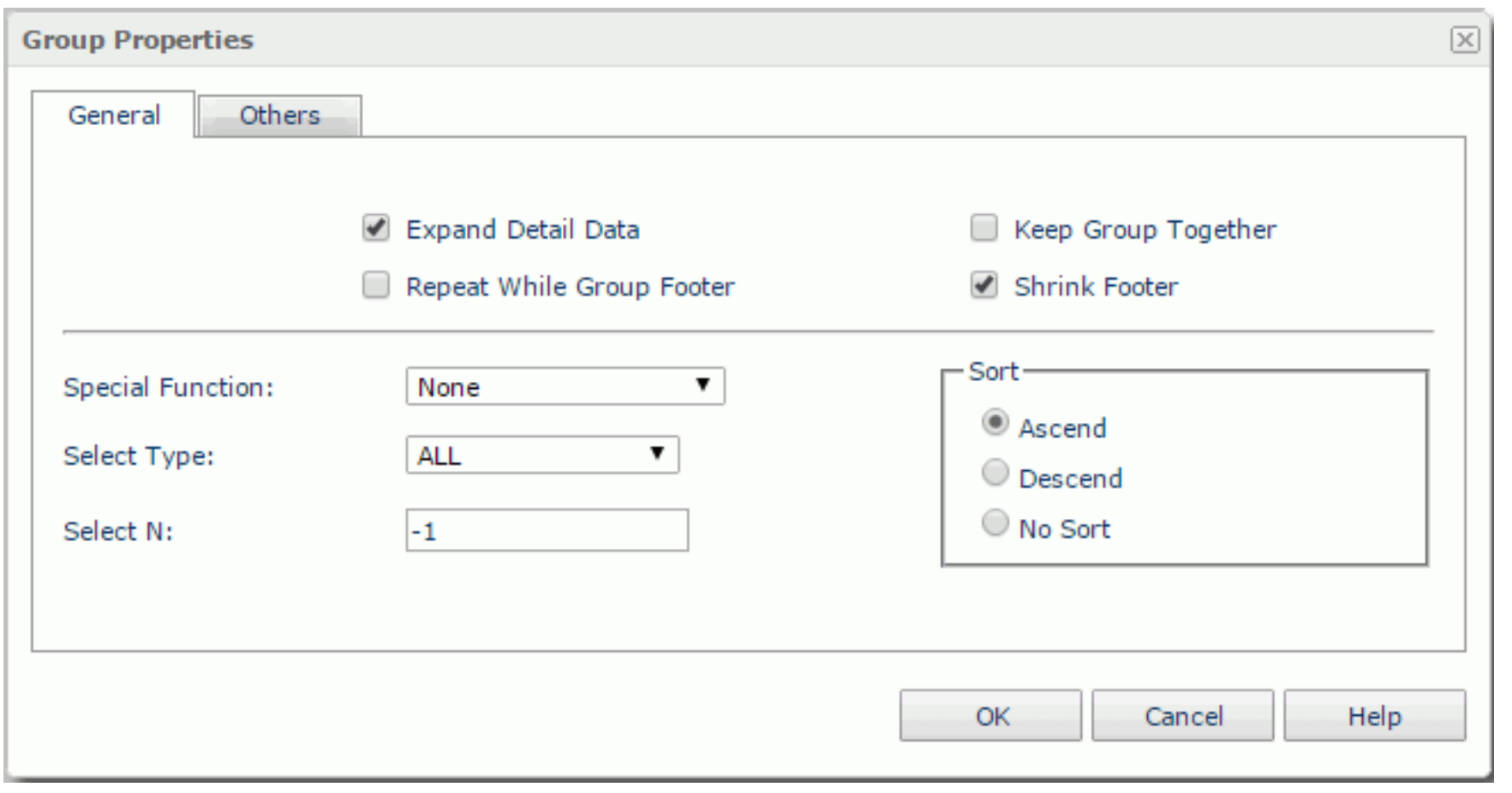
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the group object.



Expand Detail Data

Specifies whether to expand or collapse the detailed records in the group level.

Keep Group Together

Specifies whether or not to try to keep groups wholly in a report page (try not to make a group across two pages).

Repeat While Group Footer

If the group header is set to be repeated, you can specify whether or not to still repeat the group header when a page break occurs on the group footer.

Shrink Footer

Specifies whether or not to shrink the group footer when you shrink detailed records in the group level.

Special Function

Specifies a special function for the group level to group. This option is available only when the group field is of the Date/Time type.

Select Type

Specifies the type for Select N condition:

- **ALL**: The Select N condition will not take effect, and all groups will be retrieved.
- **TOP_N**: The first N groups of the group level will be displayed. Herein N is what you type in the Select N text box.
- **BOTTOM_N**: The last N groups of the group level will be displayed. N is the integer you type in the Select N text box.

Select N

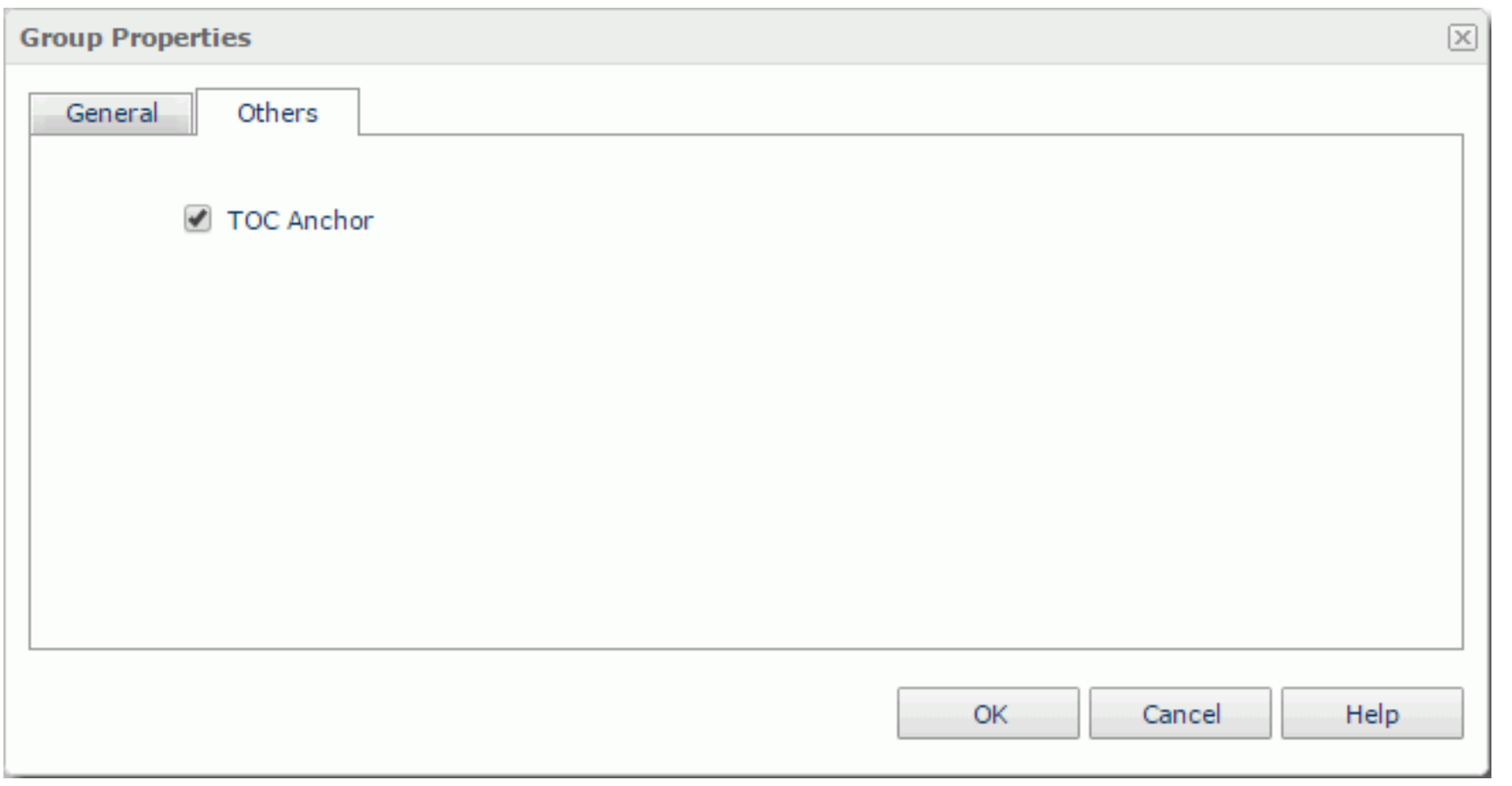
If you want only the first or last several groups of the group level to be displayed, you can type an integer which will be the number of groups to be displayed.

Sort

Specifies a sorting direction for the groups of the group level.

Others tab

You can use this tab to view and configure some miscellaneous settings.



TOC Anchor

Specifies whether or not to add the node that represents the object to the TOC tree that is displayed in the TOC Browser.

Image Properties dialog

This dialog helps you to edit the properties of an image. It contains the following tabs:

- General tab
- Others tab

OK

Applies the settings and closes this dialog.

Cancel

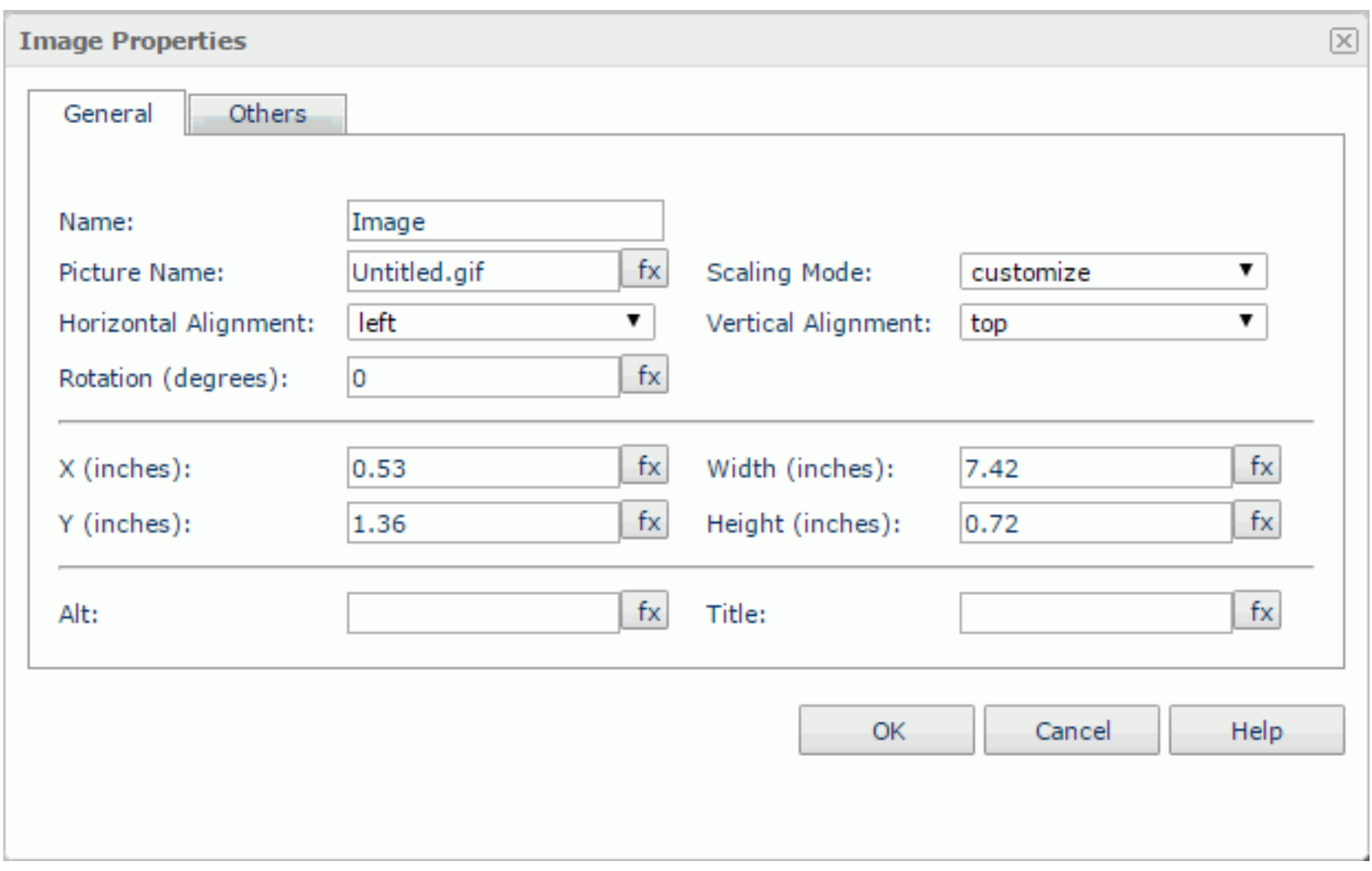
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the image.



Name

Specifies the display name of the image, which will be shown on the shortcut menu of the image.

Picture Name

Displays the image file name.

Scaling Mode

Specifies the scaling mode for the image. The mode can be:

- **Actual Size**: The image will be shown in its actual size.
- **Customize**: The image size will be equal to the image field size.
- **Fit Image**: The image will be scaled largest to wholly show in the image field.
- **Fit Width**: The image will be scaled largest to fit the width of the image field.
- **Fit Height**: The image will be scaled largest to fit the height of the image field.

Horizontal Alignment

Specifies the horizontal alignment of the image in its container.

Vertical Alignment

Specifies the vertical alignment of the image in its container.

Rotation

Rotates the image at a specified angle in degrees. The following is the meaning of different values:

- 0 - No rotation.
- Positive value - Rotate the image clockwise.
- Negative value - Rotate the image anticlockwise.

Note: When you rotate an image, the rectangle that holds the image maintains its original size, which may result in that the image exceeds the field border and therefore the parts that extend outside of the border will be cut off.

X

Specifies the X coordinate of the image.

Y

Specifies the Y coordinate of the image.

Width

Specifies the width of the image.

Height

Specifies the height of the image.

Alt

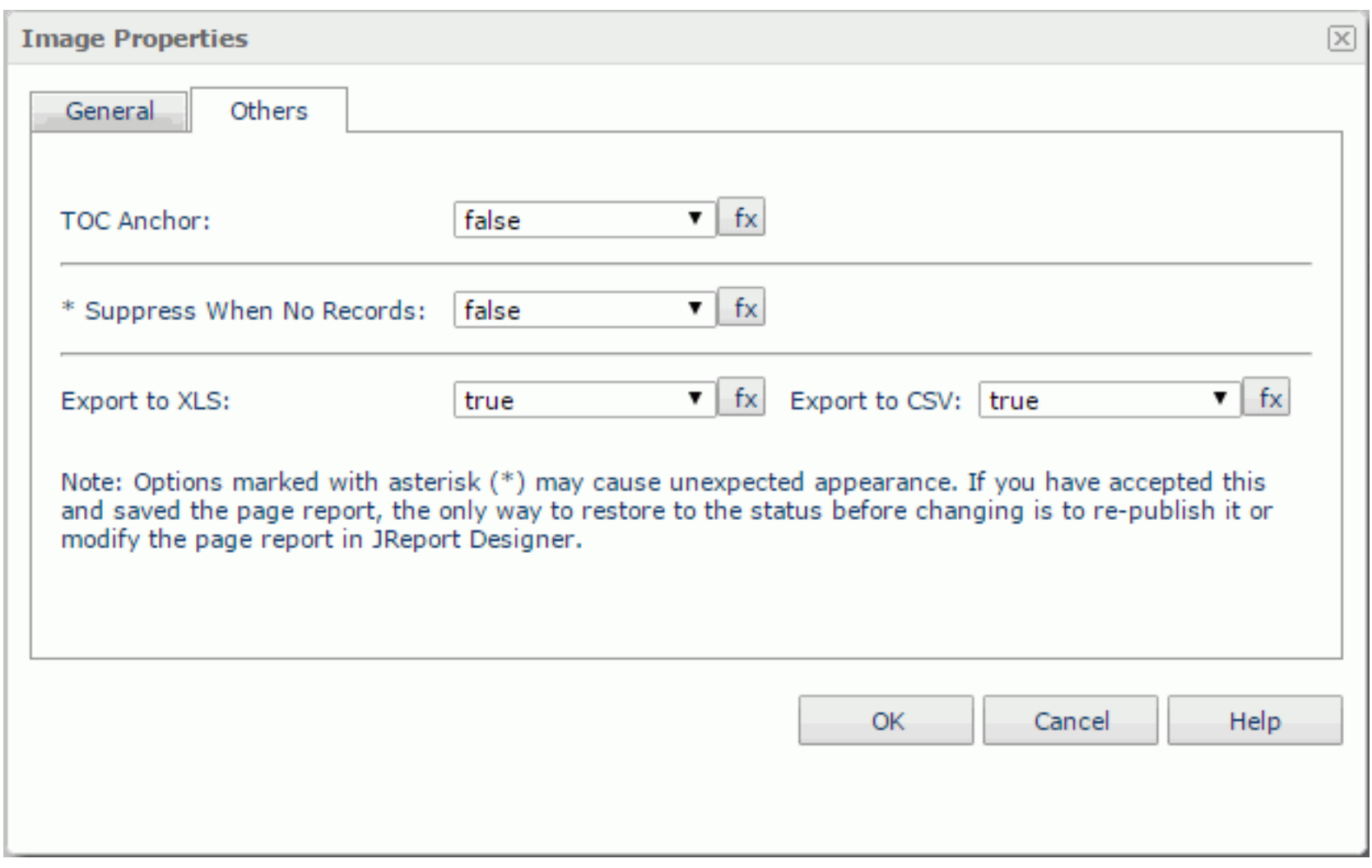
Specifies the alternate text which will be shown if the image cannot be displayed.

Title

Specifies tip information about the image, which will be displayed when the mouse cursor hovers on the image.

Others tab

You can use this tab to view and configure some miscellaneous settings.



TOC Anchor

Specifies whether or not to add the node that represents the image to the TOC tree that is displayed in the TOC Browser.

Suppress When No Records

If true and no records are returned by the report, the image will not be displayed.

Export to XLS

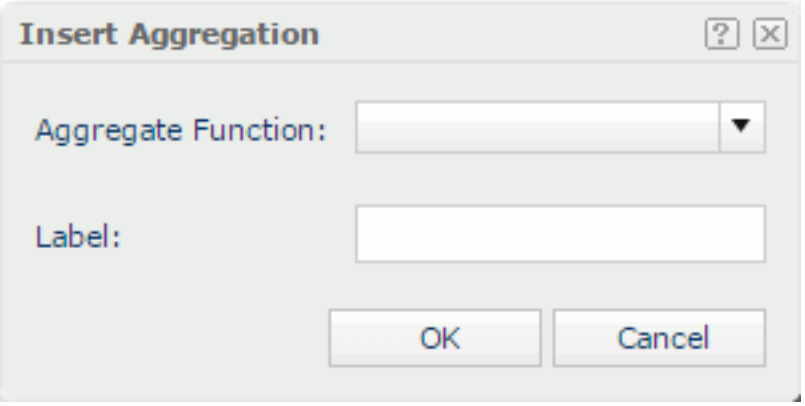
If true, the image will not be exported when you save the report result as an XLS file (make sure to check Data Format in the Export dialog).

Export to CSV

If true, the image will not be exported when you save the report result as a TXT file with Delimited Format selected.

Insert Aggregation dialog

This dialog helps you to insert the selected field as a new aggregate field in a crosstab.



Aggregate Function

Specifies the function to summarize the selected field. For details about each function, see Math functions.

If the selected field has been predefined with an aggregate function, the option is Aggregation. The name of the field is displayed in the text box and cannot be edited.

Label

Specifies the display name of the aggregate field, which will label the newly created summaries in the crosstab. You can leave it blank.

OK

Inserts the aggregation and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Help

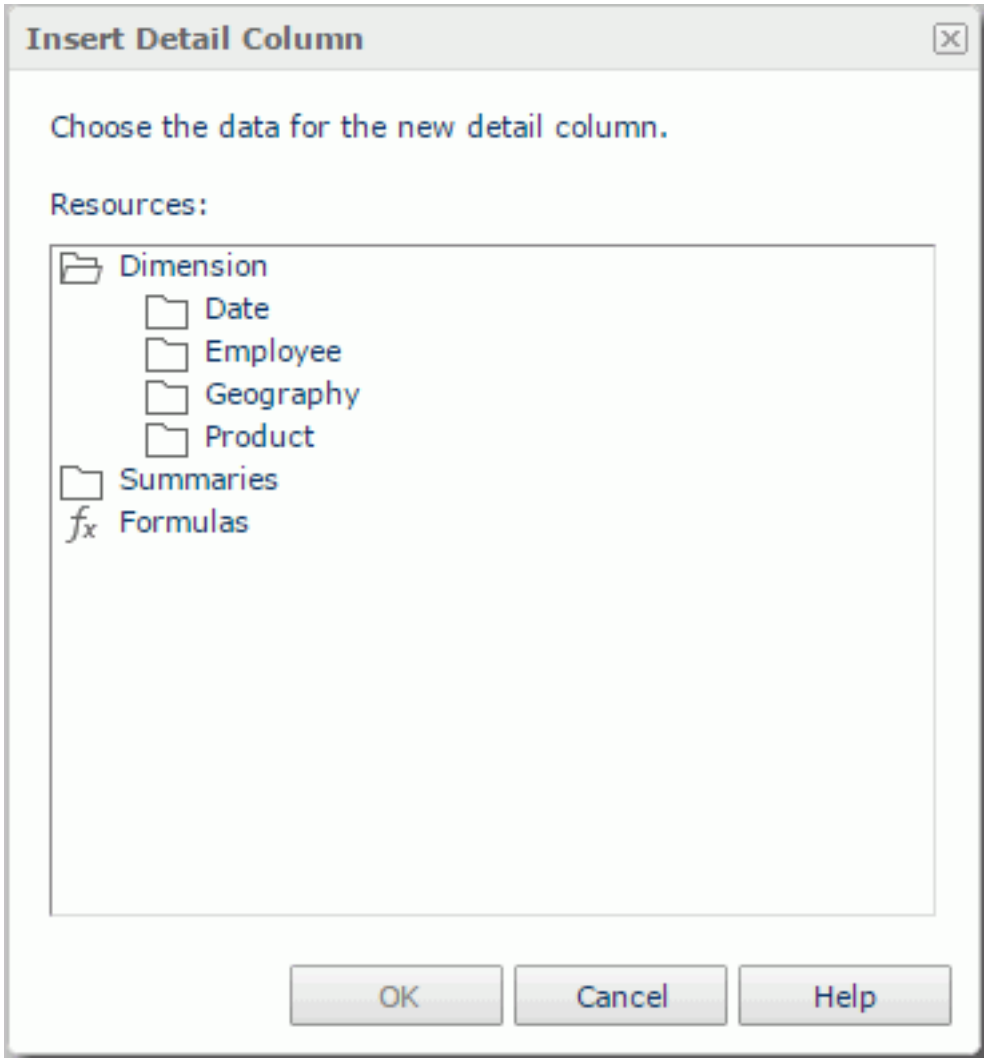
Displays the help document about this feature.

Related topics:

- Inserting aggregate fields by dragging

Insert Detail Column dialog

The dialog helps you to insert a new detail column into a table.



Resources

Lists all the available data resources for the new detail column. Select the resource you want from the list.

OK

Inserts the column with the specified resource and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Help

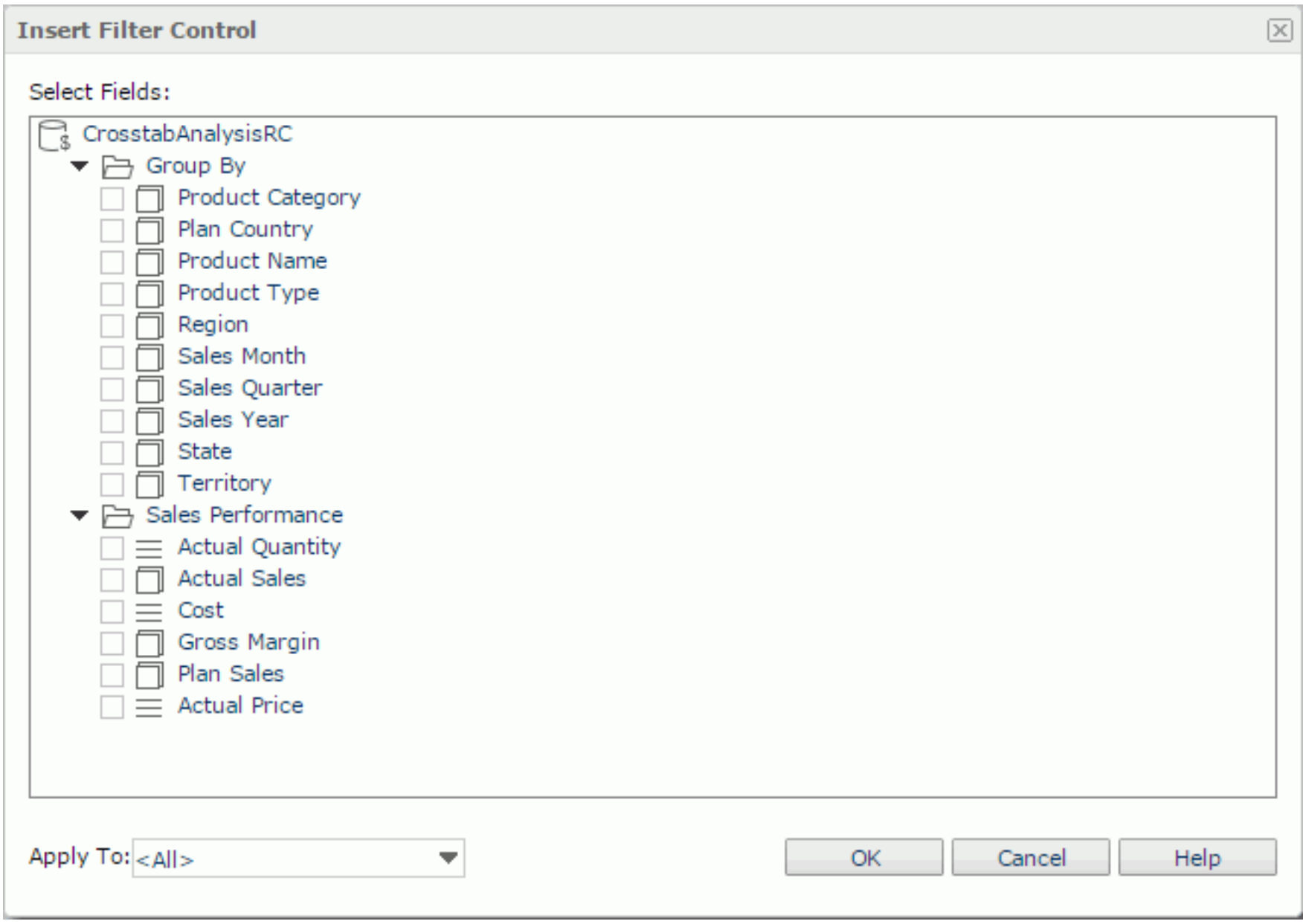
Displays the help document about this feature.

Related topics:

- Inserting table columns

Insert Filter Control dialog

The dialog helps you to insert a filter control into a report for filtering data components in the report (excluding its subreport).



Select Fields

Specifies the fields to bind to the filter control. All the selected fields should be of the same data type. All uncomparable data type fields cannot be bound to filter controls, such as Binary, Blob, Clob, Longvarchar, Longvarbinary, and Varbinary.

The common usage is to select one field, and then based on the field to filter the data of the components created from the same data source as the field.

To filter components using different data sources, choose a common field all the data sources contain and select the field in all the data sources.

Apply To

Specifies the components to which the filter created with the filter control will be applied.

OK

Inserts a filter control into the report and closes the dialog.

Cancel

Cancels the operation and closes the dialog.

Help

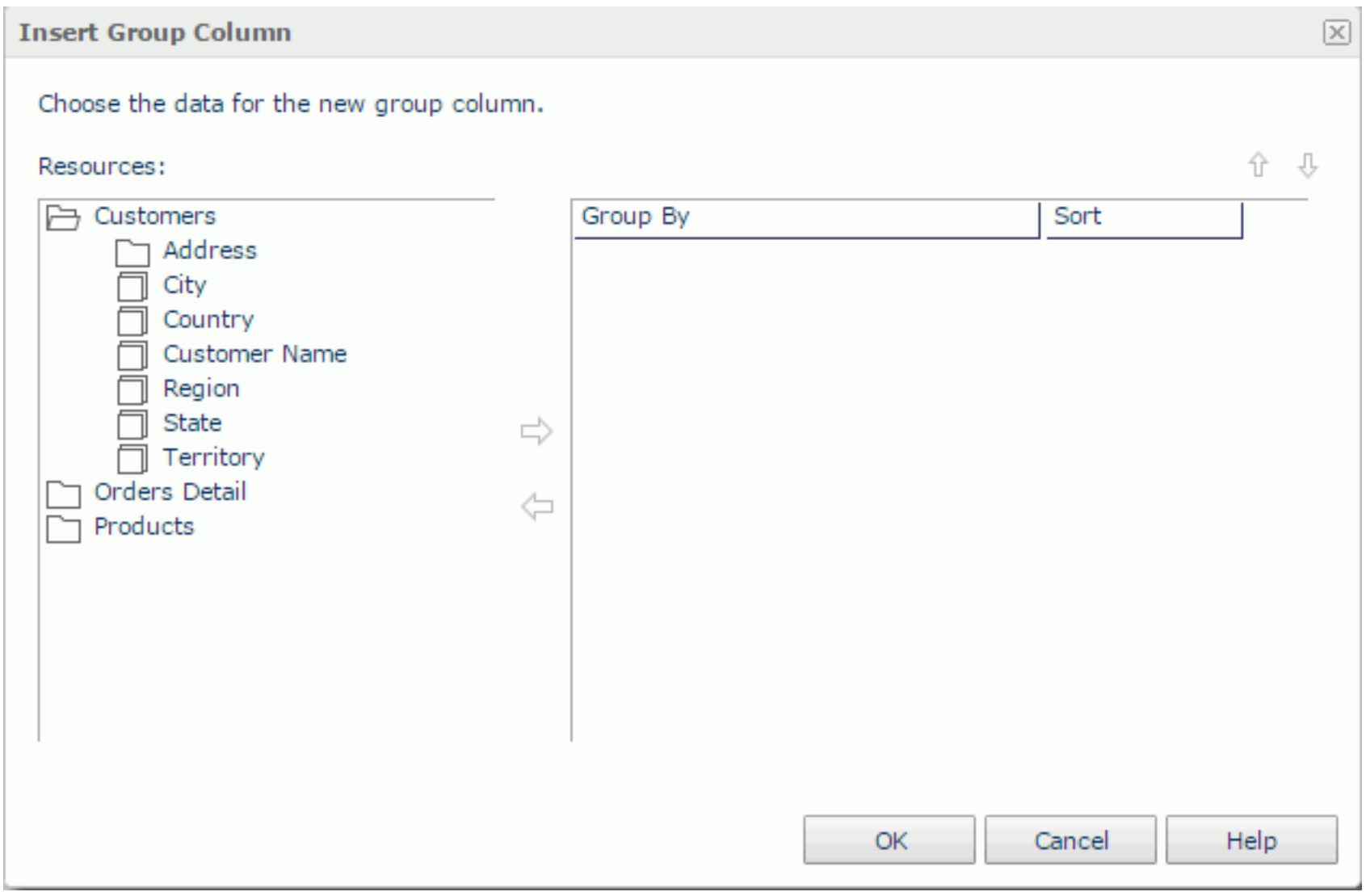
Displays the help document about this feature.

Related topics:

- Using filter control to filter report data

Insert Group Column dialog

The dialog helps you to insert new group columns into a table.



Resources

Lists all the available data resources for new group columns.

Adds the selected field to be used to group data in the table.

Removes the selected group by field.

Group By

Lists the fields that are used to group data in the table.

Sort

Specifies how groups at the specific group level will be sorted.

- **Ascend**
  Groups will be sorted in an ascending order (A, B, C).
- **Descend**
  Groups will be sorted in a descending order (C, B, A).
- **No Sort**
  Groups will be sorted in the original order in database.

Group Above

Specifies to place the selected group by field above the detail information.

Group Left Above

Specifies to place the selected group by field left above the detail information.

Group Left

Specifies to place the selected group by field left to the detail information.

Moves the selected group by field one step up.

Moves the selected group by field one step down.

OK

Inserts the group columns and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Help

Displays the help document about this feature.

Related topics:

- Inserting table columns

Insert Image dialog

This dialog helps you to insert an image into a report.

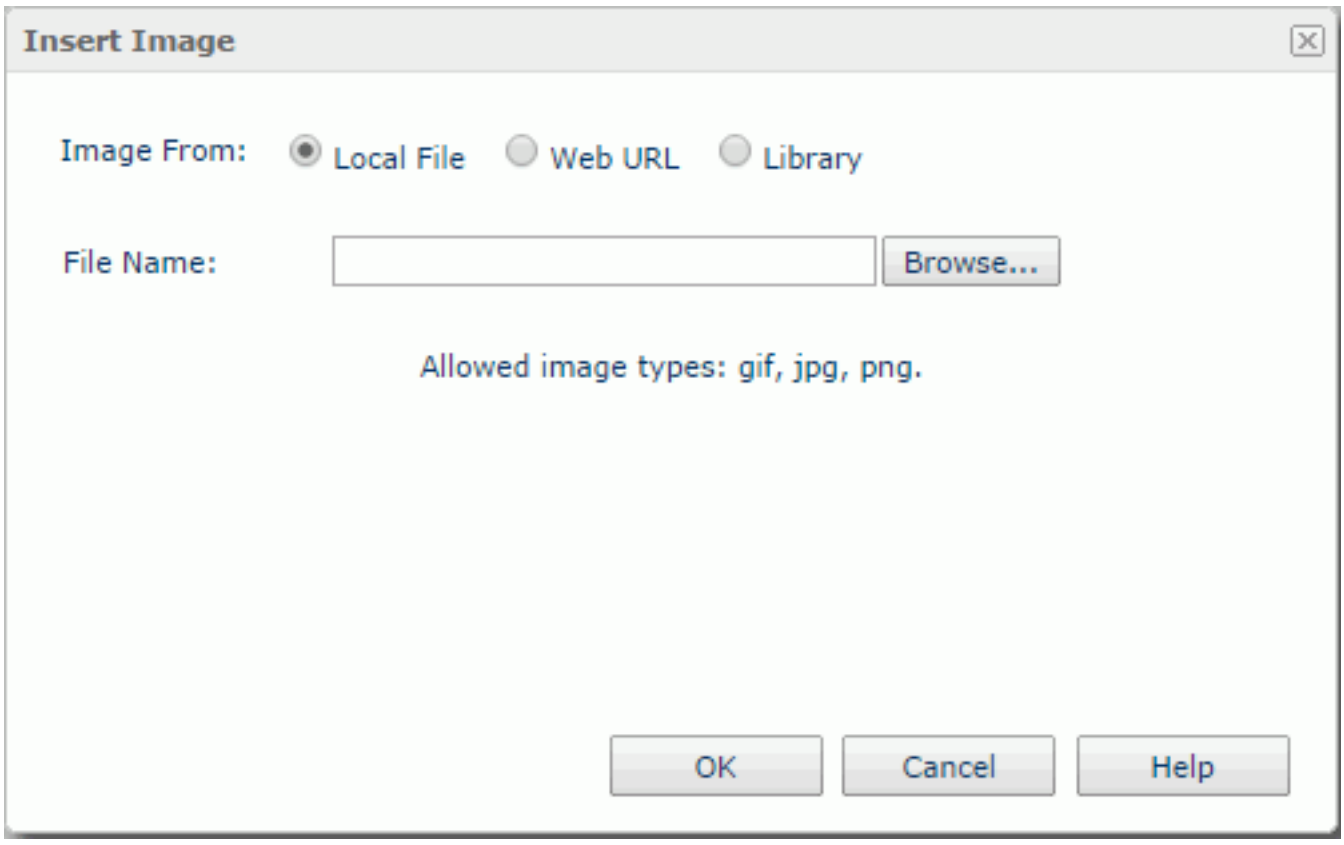


Image From

Specifies the source of the image file.

- **Local File**
  Specifies to use an image from the local file system. The types and size of the images that can be used are defined by the administrator in the Upload panel.
  - **File Name**
    Specifies the path and name of the image file. You can click the Browse button to locate the image file.
- **Web URL**
  Specifies to use an image via URL.
  - **Image URL**
    Specifies the URL of the image file. JReport will record the latest 10 entered URLs in the drop-down list.

    Note: If your JReport Server is in an intranet, to successfully access the image via URL, you need to add the parameters `-Dhttp.proxyHost=XXX -Dhttp.proxyPort=XX` to the server's startup file JRServer.bat, which locates in `<install_root>\bin`.
- **Library**
  Specifies to use an existing image.
  - **My Images**
    The My Images folder is a virtual location where JReport Server stores the images that have once been inserted into reports. Select the one you want to use.
  - **Preview**
    Displays a preview of the selected image.

OK

Inserts the image in the report and closes the dialog.

Cancel

Cancels the insertion and closes the dialog.

Help

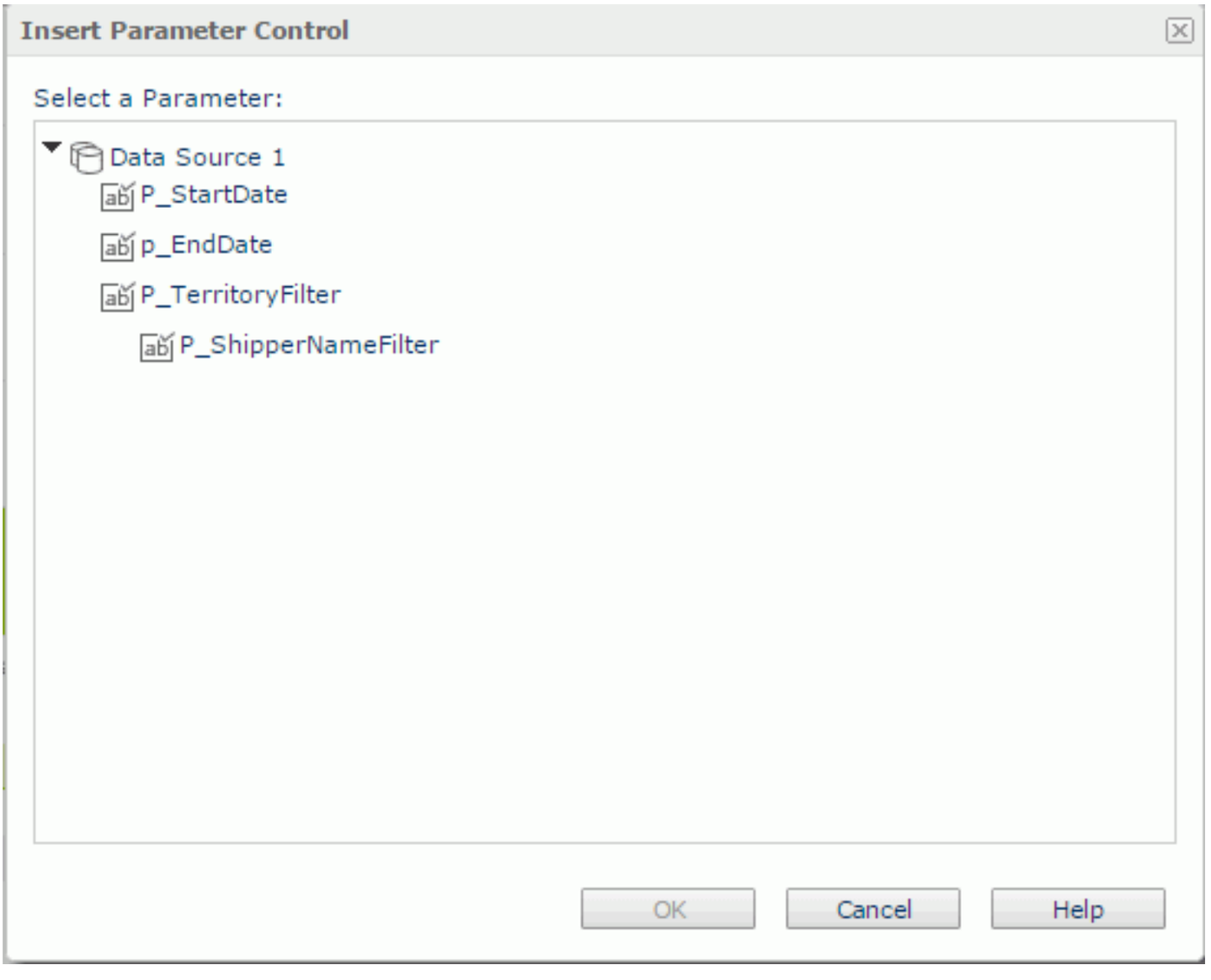
Displays the help document about this feature.

Related topics:

- Adding report objects

Insert Parameter Control dialog

The dialog helps you to insert a parameter control into a report for delivering parameter values to the report.



Select a Parameter

Specifies a parameter from the list where all the parameters except cascading parameters used by the current report are displayed.

OK

Inserts a parameter control into the report and closes the dialog.

Cancel

Cancels the operation and closes the dialog.

Help

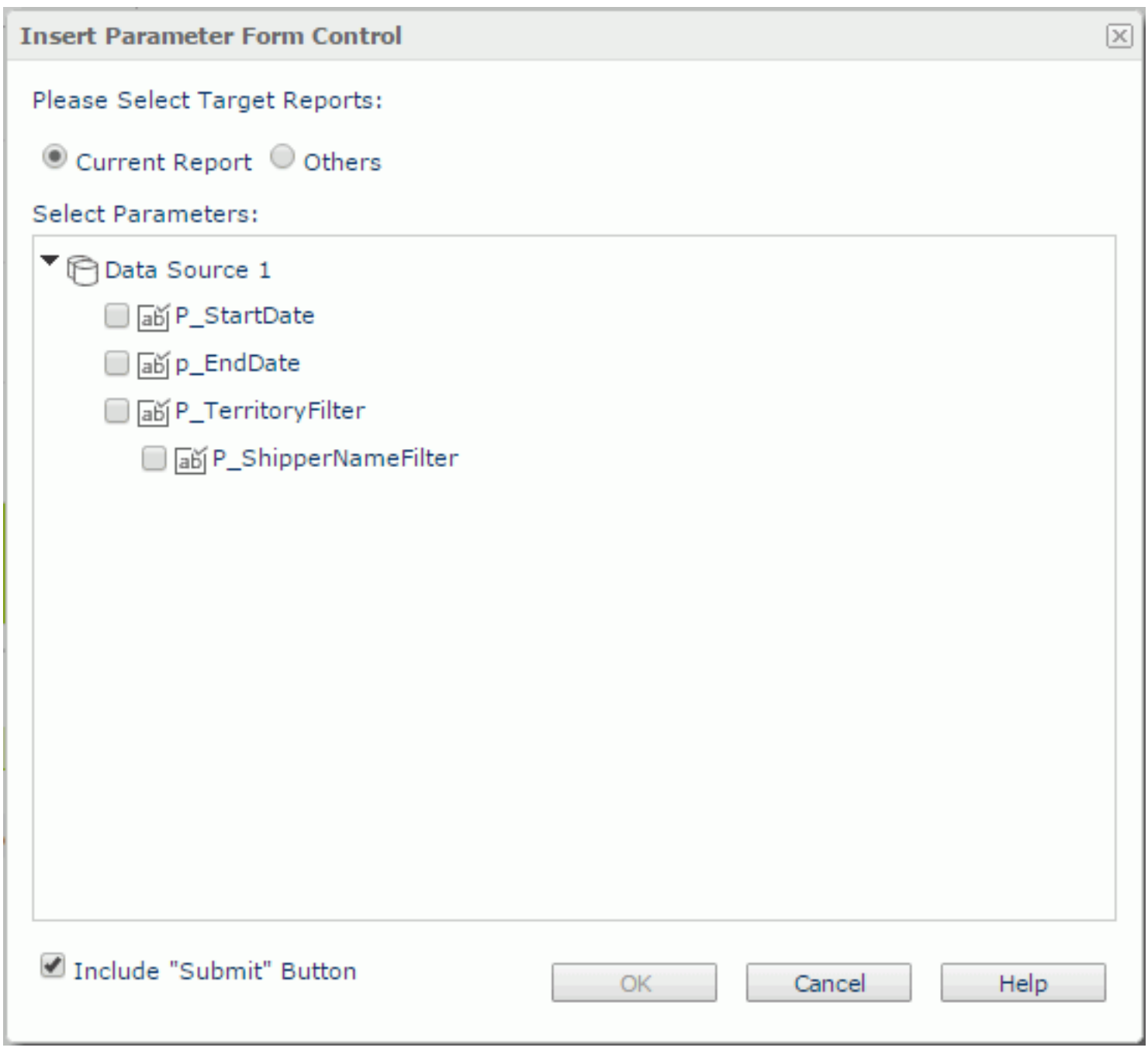
Displays the help document about this feature.

Related topics:

- Using parameter control to specify a parameter to a report

Insert Parameter Form Control dialog

The dialog helps you to insert a parameter form control into a report for running reports from the current report.



Please Select Target Reports

Specifies the reports to run.

- **Current Report**
  Specifies to run the current report with the specified parameters.
  - **Select Parameters**
    Specifies the parameters that are used to run the current report. Once you select one of a group of cascading parameters, all the other parameters in the cascading group will be selected automatically, and vice versa.
- **Others**
  Specifies to run other reports.
  - **Reports**
    Specifies the target reports to run. If all the selected reports contain no parameters, you cannot finish the dialog.
  - **Parameter Information**
    You can click a report and then view its parameters here.

Include "Submit" Button

Specifies whether the inserted parameter form control contains the Submit button. If Submit is included, it is used to submit the parameter values you specified in the parameter form control. If Submit is not included, once you change the values of a parameter in the parameter form control, the new values will be applied automatically.

OK

Inserts a parameter form control into the report and closes the dialog.

Cancel

Cancels the operation and closes the dialog.

Help

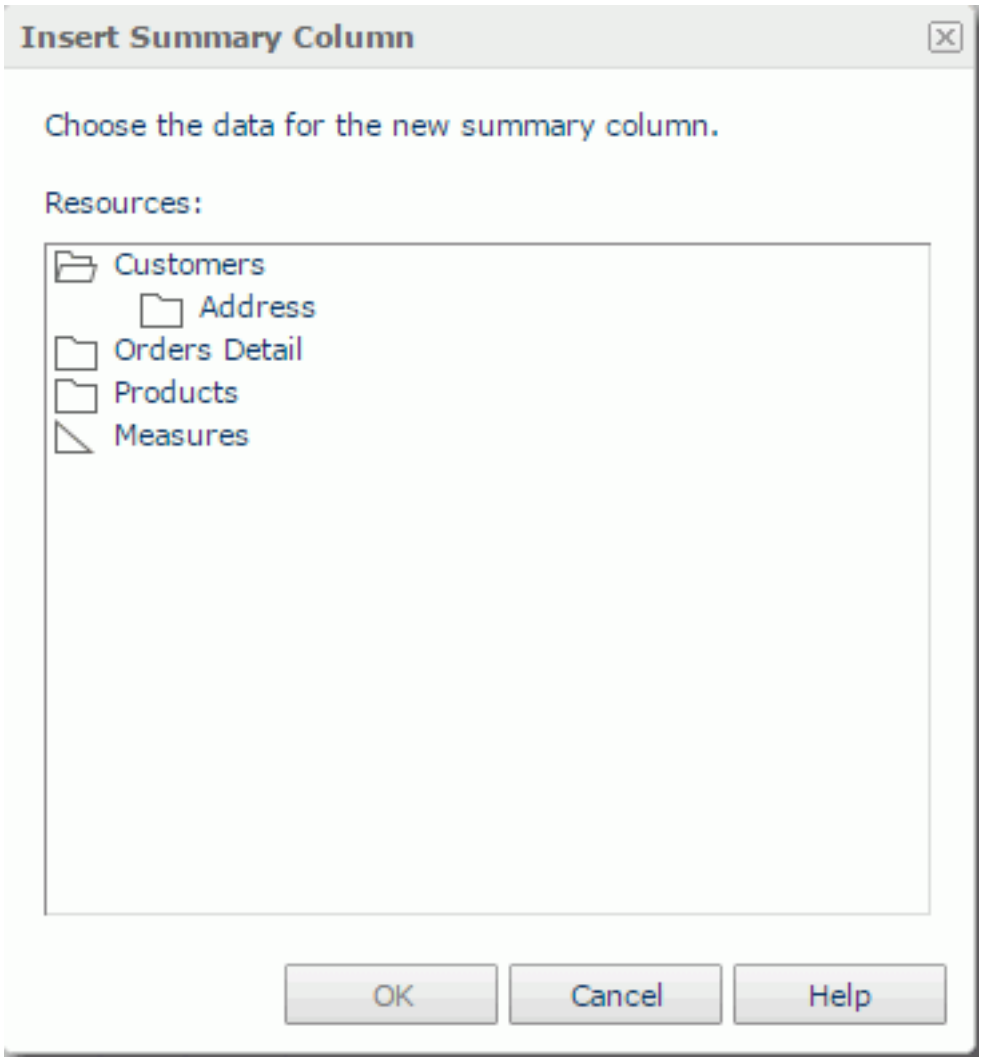
Displays the help document about this feature.

Related topics:

- Using parameter form control to run reports

Insert Summary Column dialog

The dialog helps you to insert a new summary column into a table.



Resources

Lists all the available data resources for the new summary column. Select the resource you want from the list.

OK

Inserts the column with the specified resource and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Help

Displays the help document about this feature.

Related topics:

- Inserting table columns

Label Properties dialog

This dialog helps you to edit the properties of a label. It contains the following tabs:

- General tab
- Font tab
- Border tab
- Others tab
- Display tab

OK

Applies the settings and closes this dialog.

Cancel

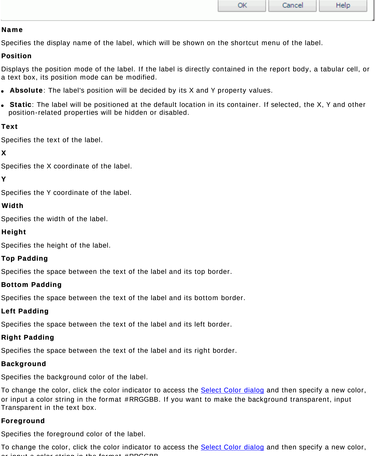
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the label.



Name

Specifies the display name of the label, which will be shown on the shortcut menu of the label.

Position

Displays the position mode of the label. If the label is directly contained in the report body, a tabular cell, or a text box, its position mode can be modified.

- **Absolute**: The label's position will be decided by its X and Y property values.
- **Static**: The label will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled.

Text

Specifies the text of the label.

X

Specifies the X coordinate of the label.

Y

Specifies the Y coordinate of the label.

Width

Specifies the width of the label.

Height

Specifies the height of the label.

Top Padding

Specifies the space between the text of the label and its top border.

Bottom Padding

Specifies the space between the text of the label and its bottom border.

Left Padding

Specifies the space between the text of the label and its left border.

Right Padding

Specifies the space between the text of the label and its right border.

Background

Specifies the background color of the label.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

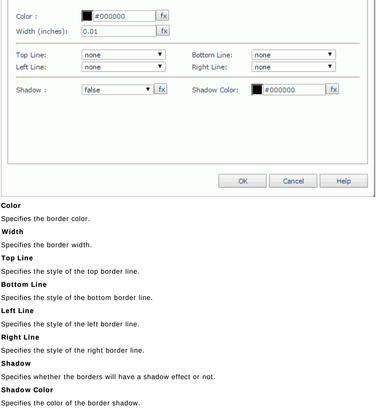
Foreground

Specifies the foreground color of the label.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB.

Font tab

This tab shows the font-related information of the label.



Font

Specifies the font face of the label text.

Size

Specifies the font size of the label text.

Horizontal Alignment

Specifies the horizontal alignment mode of the text in the label.

Vertical Alignment

Specifies the vertical alignment mode of the text in the label.

Bold

Specifies whether to make the label text bold or not.

Underline

Specifies whether the label text will be underlined or not.

Strikethrough

Specifies whether or not to attach a strikeout line to the label text.

Italic

Specifies whether to make the label text italic or not.

Autofit

Specifies whether or not to automatically adjust the width of the label.

Word Wrap

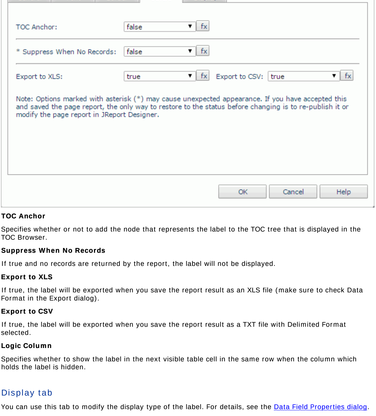
Specifies whether or not to wrap the text to the label width.

Ignore HTML Tag

If this option is unchecked, JReport will parse HTML tag elements in the field value while the report is to be saved as an HTML file; or the field value will appear in the HTML file the same as that in Page Report Studio (HTML tag elements in the field value, if any, will not be parsed).

Border tab

This tab shows information about borders of the label. You can modify all the border settings in this tab.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Shadow

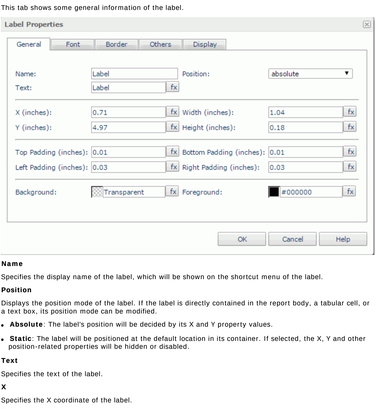
Specifies whether the borders will have a shadow effect or not.

Shadow Color

Specifies the color of the border shadow.

Others tab

You can use this tab to view and configure some miscellaneous settings.



TOC Anchor

Specifies whether or not to add the node that represents the label to the TOC tree that is displayed in the TOC Browser.

Suppress When No Records

If true and no records are returned by the report, the label will not be displayed.

Export to XLS

If true, the label will be exported when you save the report result as an XLS file (make sure to check Data Format in the Export dialog).

Export to CSV

If true, the label will be exported when you save the report result as a TXT file with Delimited Format selected.

Logic Column

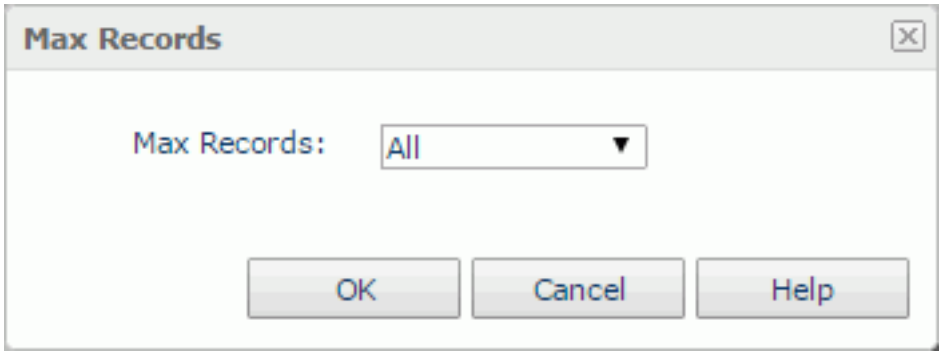
Specifies whether to show the label in the next visible table cell in the same row when the column which holds the label is hidden.

Display tab

You can use this tab to modify the display type of the label. For details, see the Data Field Properties dialog.

Max Records dialog

This dialog allows you to set the maximum number of records for components to retrieve.



Max Records

Specifies the maximum record number. All means that all records will be retrieved, Top 10/20/50/100 means that the first 10/20/50/100 records will be retrieved, and you can directly input a number here to retrieve the corresponding first records.

Note: If any filter condition has been set on the component, which will filter out some records, the number of retrieved records may not accord with the number set for Max Records.

OK

Applies the settings to report components and closes this dialog.

Cancel

Discards the settings and closes this dialog.

Help

Displays the help document about this feature.

Multivalue Container Properties dialog

This dialog helps you to edit the properties of a multivalue container. It contains the following tabs:

- General tab
- Font tab
- Border tab
- Others tab

OK

Applies the settings and closes this dialog.

Cancel

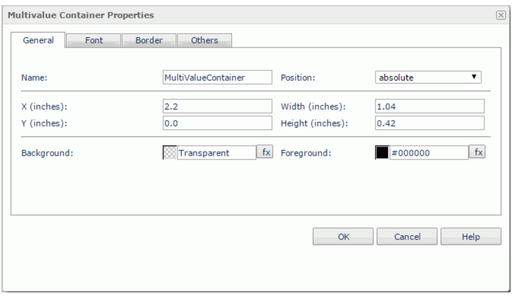
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the multivalue container.



Name

Specifies the display name of the multivalue container, which will be shown on its shortcut menu.

Position

Specifies the position mode of the multivalue container. If the multivalue container is directly contained in the report body, a tabular cell, or a text box, its position mode can be modified.

- **Absolute**: The multivalue container's position will be decided by its X and Y property values.
- **Static**: The multivalue container will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled.

X

Specifies the X coordinate of the multivalue container.

Y

Specifies the Y coordinate of the multivalue container.

Width

Specifies the width of the multivalue container.

Height

Specifies the height of the multivalue container.

Background

Specifies the background color of the multivalue container.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

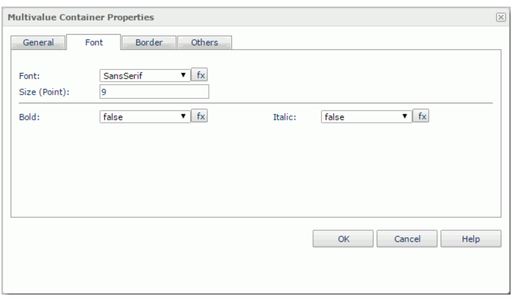
Foreground

Specifies the foreground color of the multivalue container.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the text box.

Font tab

This tab shows the font-related information of the multivalue container.



Font

Specifies the font face of the text.

Size

Specifies the font size of the text.

Bold

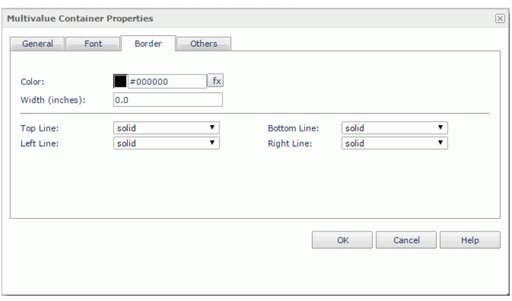
Specifies whether to make the text bold or not.

Italic

Specifies whether to make the text italic or not.

Border tab

This tab shows information about borders of the multivalue container.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

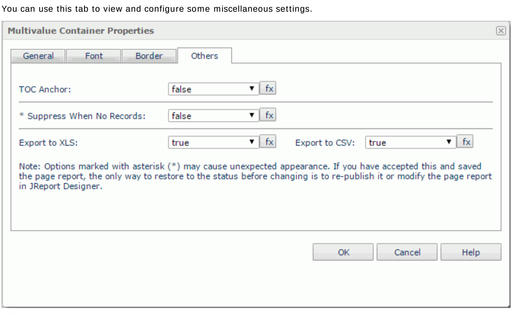
Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Others tab

You can use this tab to view and configure some miscellaneous settings.



TOC Anchor

Specifies whether or not to add the node that represents the multivalue container to the TOC tree that is displayed in the TOC Browser.

Suppress When No Records

If true and no records are returned by the report, the multivalue container will not be displayed.

Export to XLS

If true, the multivalue container will be exported when you save the report result as an XLS file (make sure to check Data Format in the Export dialog).

Export to CSV

If true, the multivalue container will be exported when you save the report result as a TXT file with Delimited Format selected.

Navigation Control Properties dialog

This dialog helps you to edit the properties of a navigation control. It contains the following tabs:

- General tab
- Border tab

OK

Applies the settings and closes this dialog.

Cancel

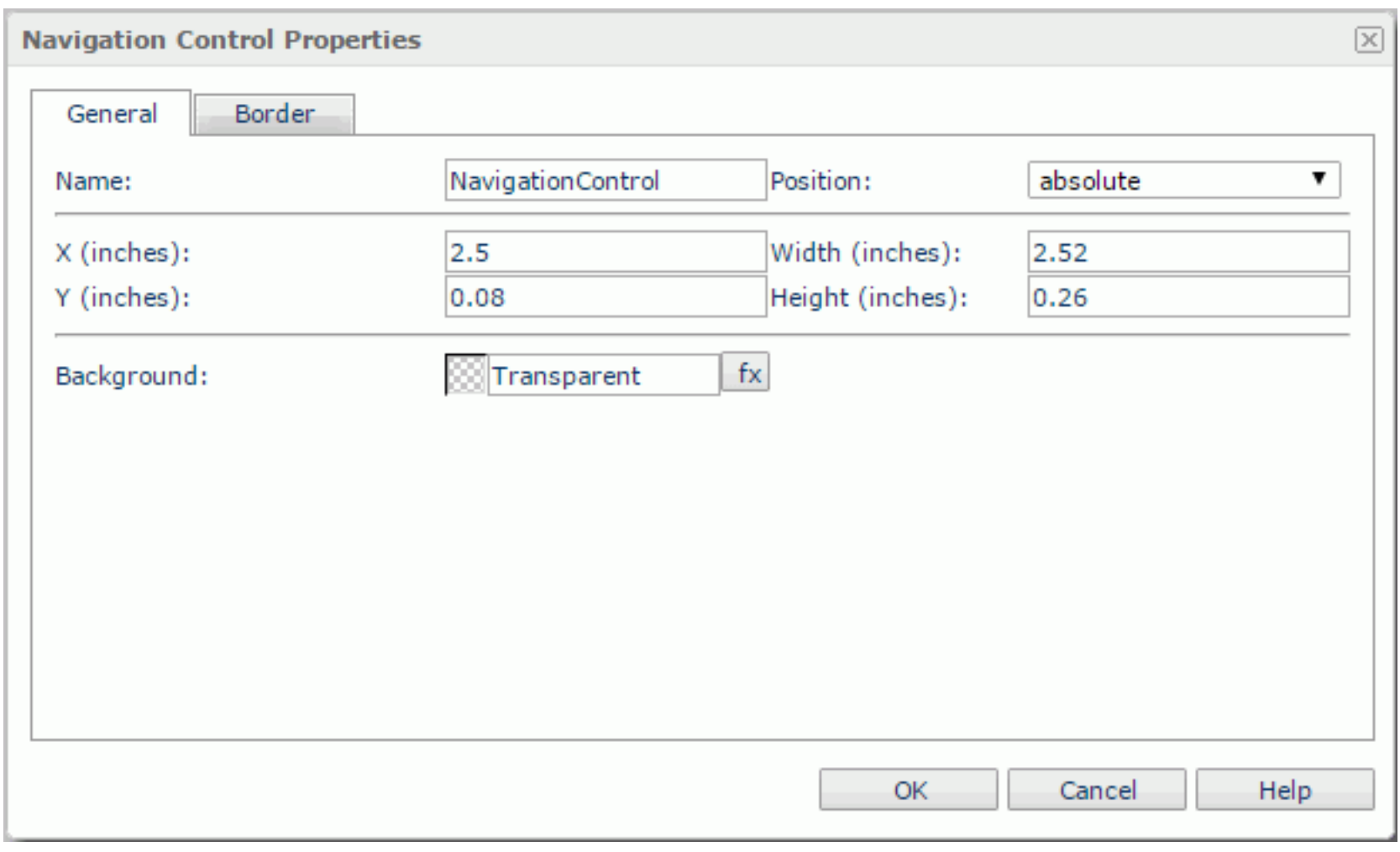
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the navigation control.



Name

Specifies the name of the navigation control.

Position

Displays the position mode of the navigation control. If the navigation control is directly contained in the report body, a tabular cell, or a text box, its position mode can be modified.

- **Absolute**: The navigation control's position will be decided by its X and Y property values.
- **Static**: The navigation control will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled.

X

Specifies the X coordinate of the navigation control.

Y

Specifies the Y coordinate of the navigation control.

Width

Specifies the width of the navigation control.

Height

Specifies the height of the navigation control.

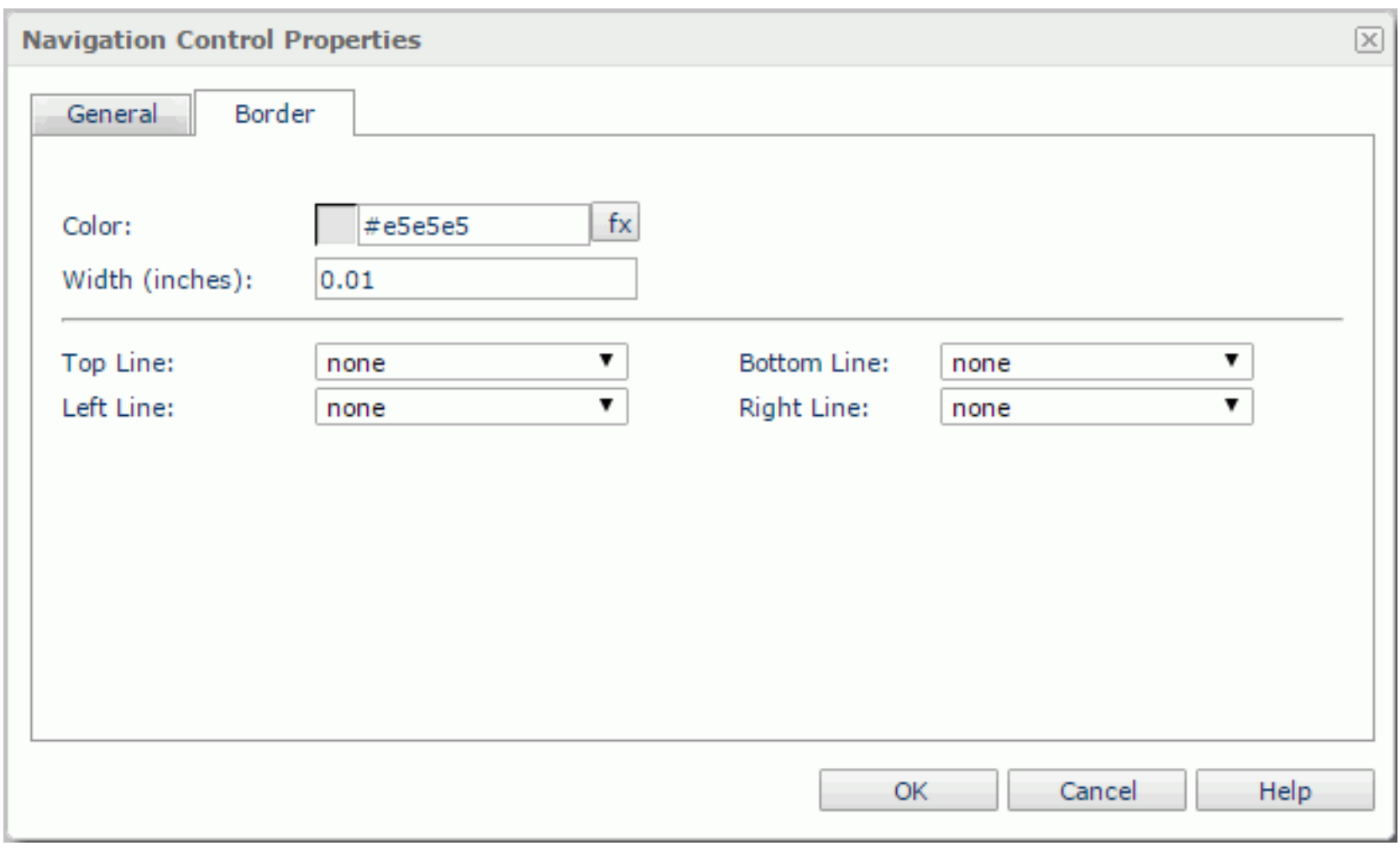
Background

Specifies the background color of the navigation control.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the navigation control.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

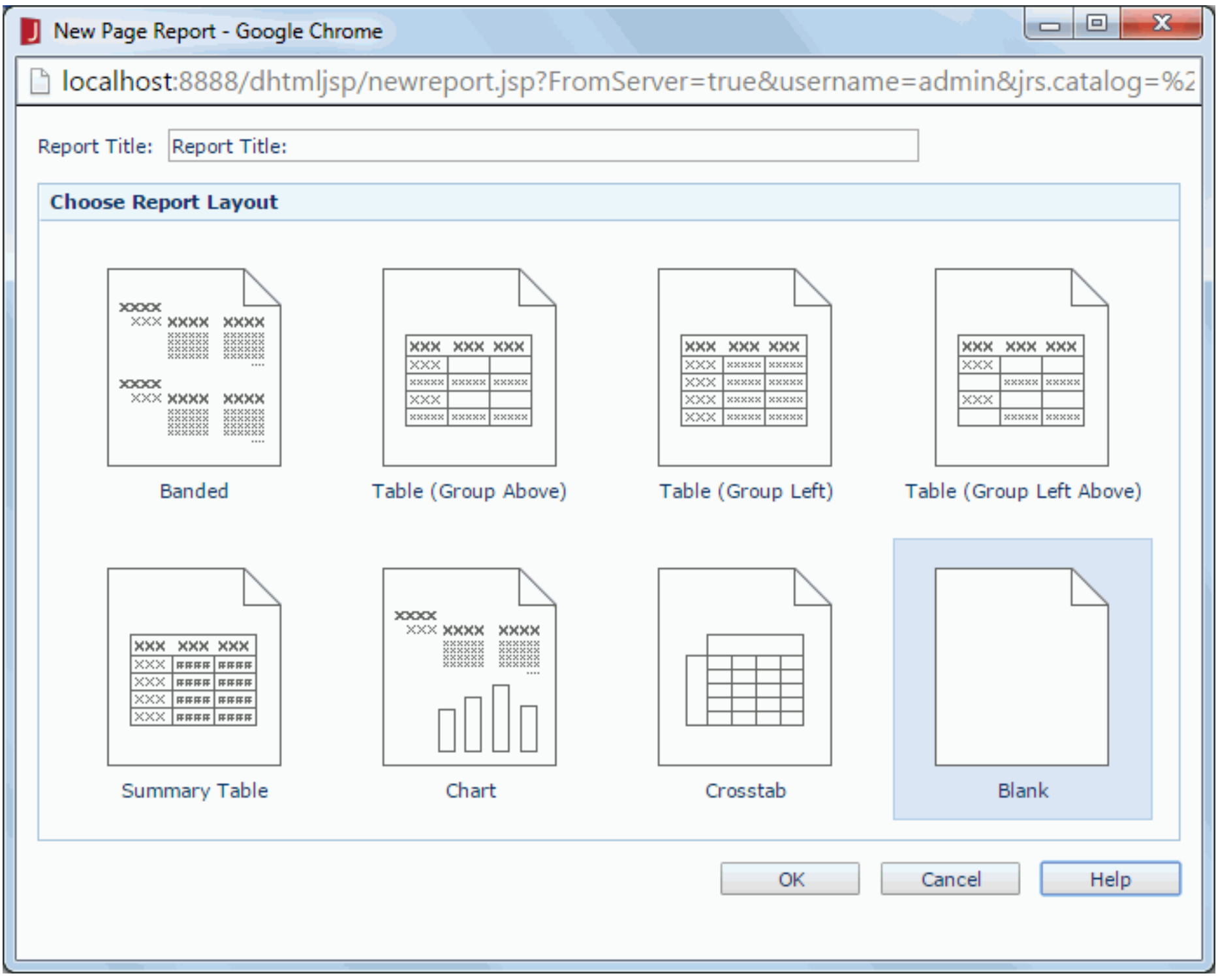
Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

New Page Report dialog

This dialog helps you to create a new page report that contains one report tab.



Report Title

Specifies the title of the report tab.

Choose Report Layout

Lists the report layouts on which you can create your report tab.

- **Banded**
  Creates a report containing a banded object.
- **Table (Group Above)**
  Creates a report containing a table with group information above the detail panel.
- **Table (Group Left)**
  Creates a report containing a table with group information left to the detail panel.
- **Table (Group Left Above)**
  Creates a report containing a table with group information left above the detail panel.
- **Summary Table**
  Creates a report containing a table with only group and summary information.
- **Chart**
  Creates a report containing a chart.
- **Crosstab**
  Creates a report containing a crosstab.
- **Blank**
  Creates a report with nothing in it.

OK

Closes this dialog, and in accord with the selected report layout, the corresponding report wizard will be displayed, or a new page report will be created with a blank report tab if Blank is selected.

- Banded Wizard
- Table Wizard
- Chart Wizard
- Crosstab Wizard

Cancel

Closes this dialog without creating a page report.

Help

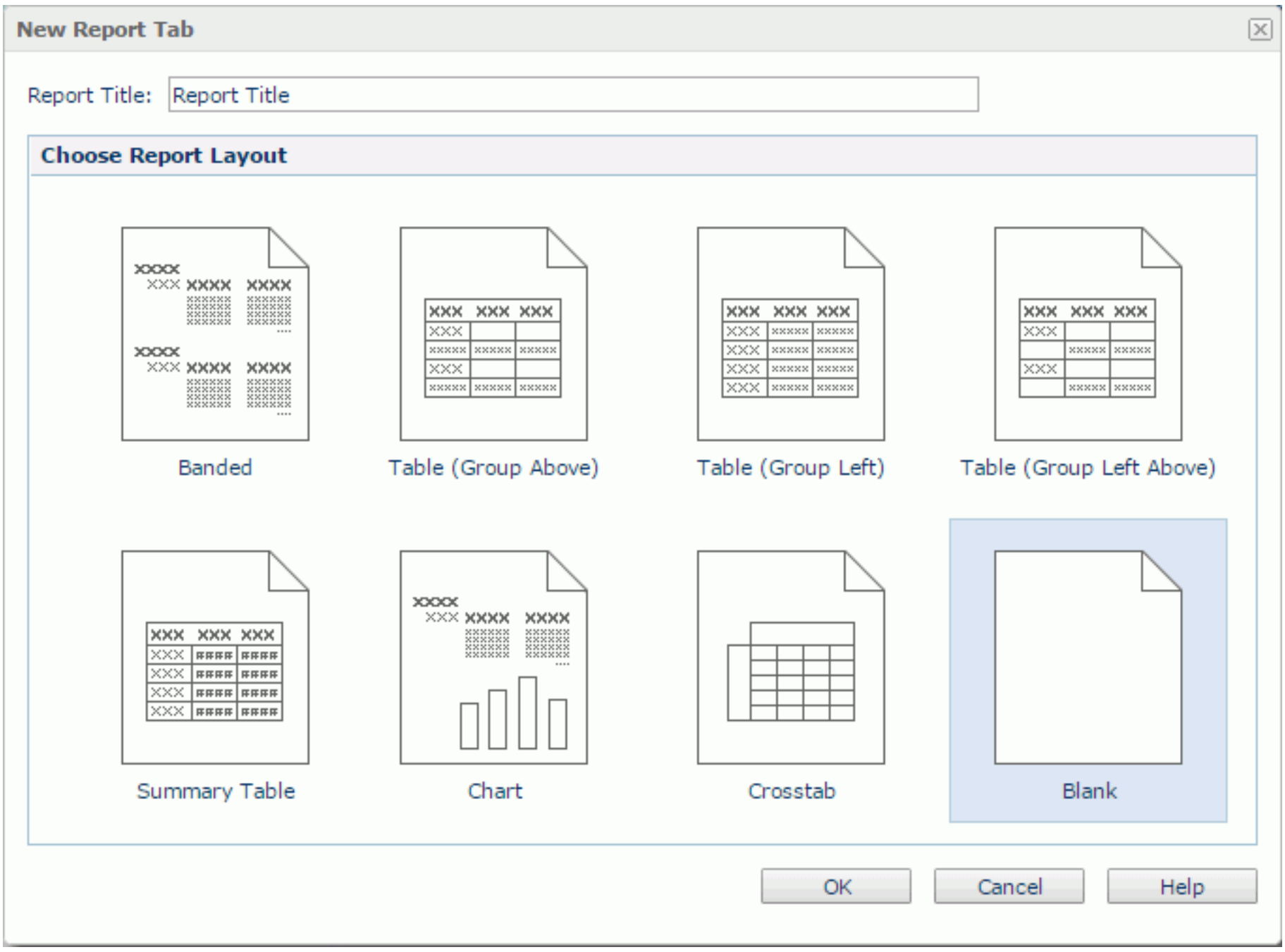
Displays the help document about this feature.

Related topics:

- Creating a report tab

New Report Tab dialog

This dialog helps you to append a new report tab to the current report.



Report Title

Specifies the title of the report tab.

Choose Report Layout

Lists the report layouts on which you can create your report tab.

- **Banded**
  Creates a report containing a banded object.
- **Table (Group Above)**
  Creates a report containing a table with group information above the detail panel.
- **Table (Group Left)**
  Creates a report containing a table with group information left to the detail panel.
- **Table (Group Left Above)**
  Creates a report containing a table with group information left above the detail panel.
- **Summary Table**
  Creates a report containing a table with only group and summary information.
- **Chart**
  Creates a report containing a chart.
- **Crosstab**
  Creates a report containing a crosstab.
- **Blank**
  Creates a report with nothing in it.

OK

Closes this dialog, and in accord with the selected report layout, the corresponding report wizard will be displayed, or a blank report will be created if Blank is selected.

- Banded Wizard
- Table Wizard
- Chart Wizard
- Crosstab Wizard

Cancel

Closes this dialog without creating a report tab.

Help

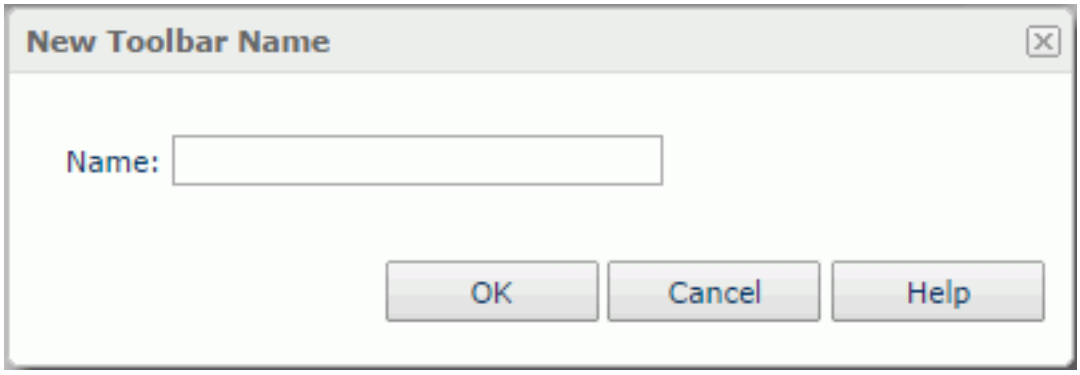
Displays the help document about this feature.

Related topics:

- Creating a report tab

New Toolbar Name dialog

This dialog helps you to input the name for the new toolbar.



Name

Specifies the name of the new toolbar.

A valid toolbar name:

- Can only contain alphanumeric characters, underlines and spaces (except for at the beginning or end of the name).
- Cannot be the same as that of an existing toolbar.

OK

Accepts a valid name (see above), and returns to the Options dialog.

Cancel

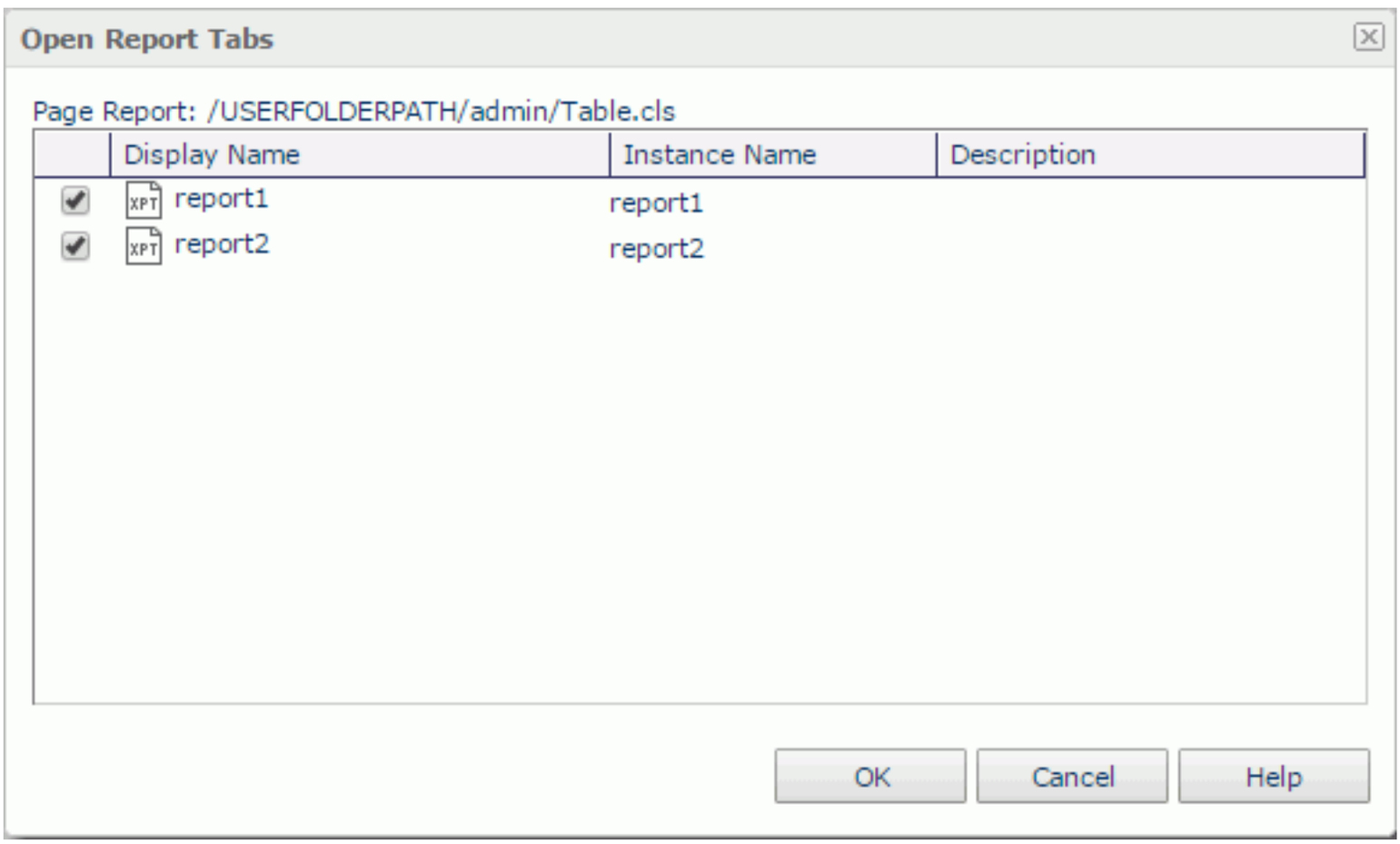
Closes this dialog without creating a new toolbar.

Help

Displays the help document about this feature.

Open Report Tabs dialog

This dialog helps you to open or hide report tabs in the current report, and shows the display name and description of each report tab.



In this dialog, the open report tabs in the current report are marked with a check symbol. To change the display state of the report tabs in the report, check those report tabs to be opened, uncheck those to be closed, and then click the OK button.

Display Name

Shows the display names of the report tabs in the report.

Instance Name

Shows the instance names of the report tabs in the report.

Description

Shows the descriptions of the report tabs if any.

OK

Closes this dialog, opens the selected report tabs, and closes those deselected.

Cancel

Closes this dialog without changing the report display state.

Help

Displays the help document about this feature.

Options dialog

This dialog helps you to set the user interface of Page Report Studio. It contains the following tabs:

- Option tab
- Customize tab

Restore Defaults

Restores the default settings.

OK

Applies the settings and closes the Options dialog.

Cancel

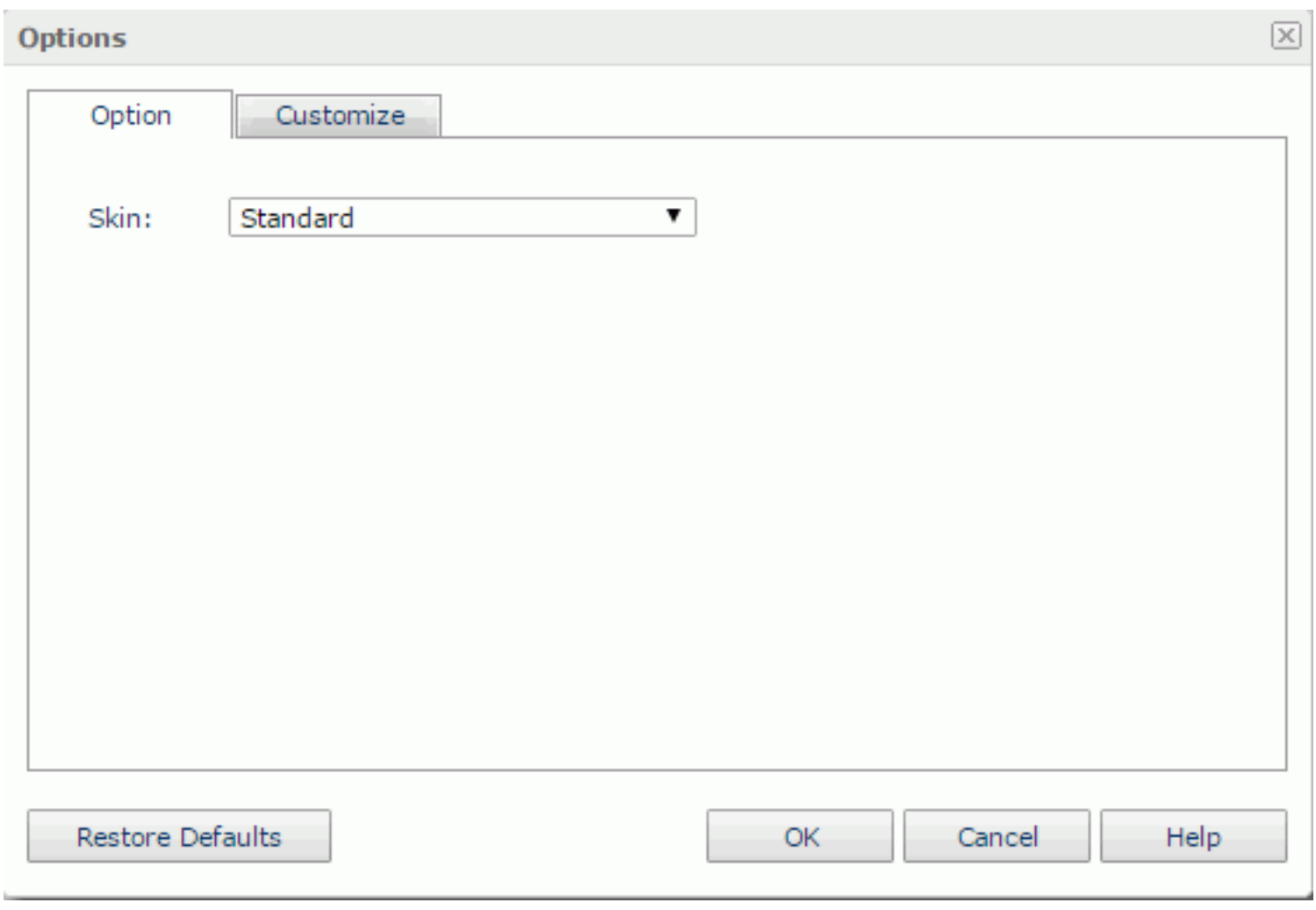
Cancels the settings and closes the Options dialog.

Help

Displays the help document about this feature.

Option tab

This tab allows you to specify the skin of Page Report Studio user interface.

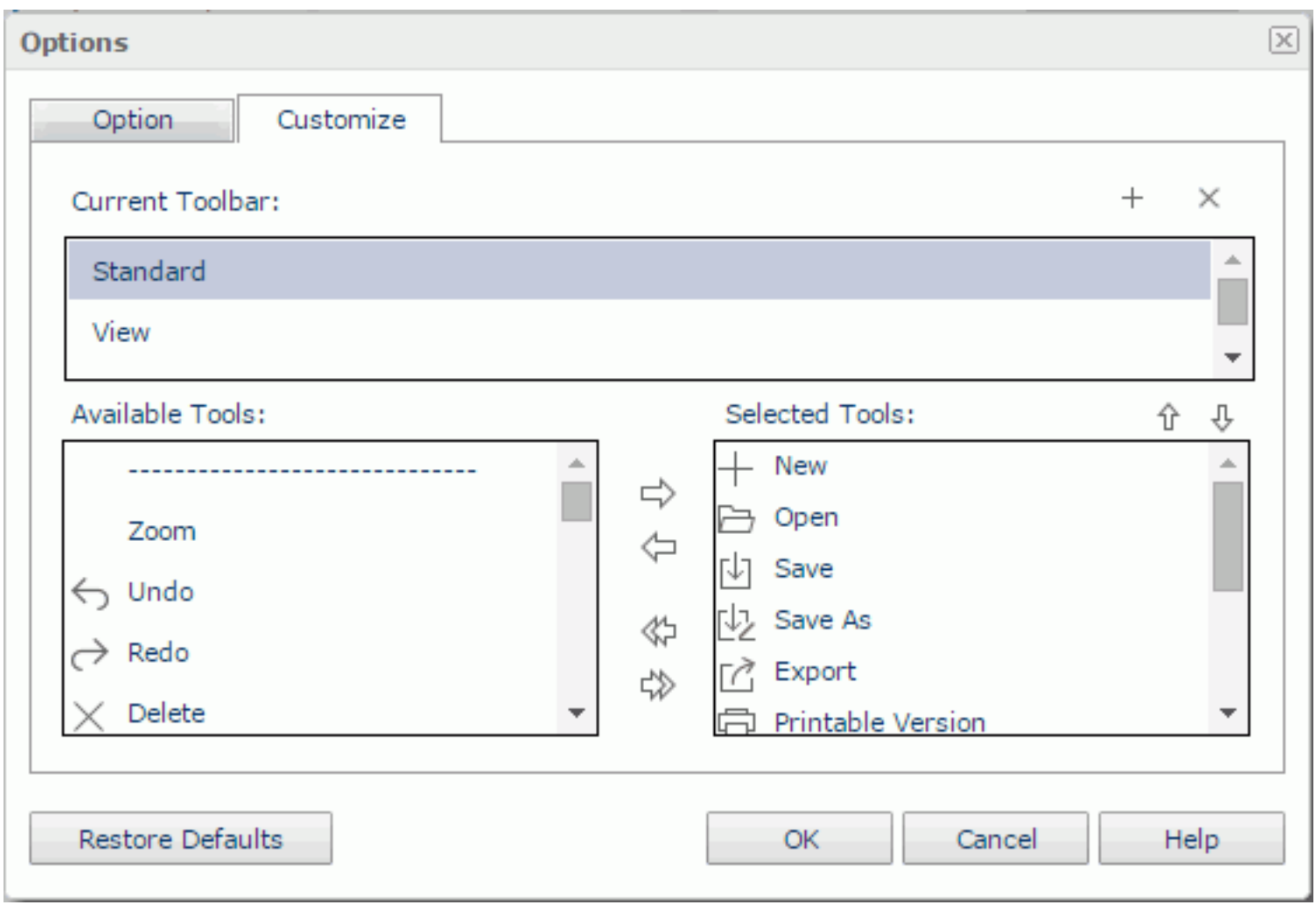


Skin

Specifies the skin of the Page Report Studio user interface. The skin can be the Standard, Classical, or Windows XP style.

Customize tab

This tab allows you to customize the toolbars in Page Report Studio.



Current Toolbar

Lists all the available toolbars. If you want to customize a toolbar, click it to select.

Displays the New Toolbar Name dialog for you to input the name for the new toolbar. The new toolbar name will then be displayed in the Current Toolbar list box.

Removes the selected toolbar.

Available Tools

Displays the tools that you can add to the toolbars. The tools available in the box are determined by which feature profile is applied.

Selected Tools

Displays the tools of the toolbar selected in the Current Toolbar list box in the order that they will appear (from left to the right) on the toolbar.

Adds the selected tool from the Available Tools list box to the Selected Tools list box.

Removes the selected tool.

Empties the contents of the Selected Tools list box.

Adds all the available tools from the Available Tools list box to the Selected Tools list box.

Adjusts the location of the selected tool by moving it to the left on the toolbar.

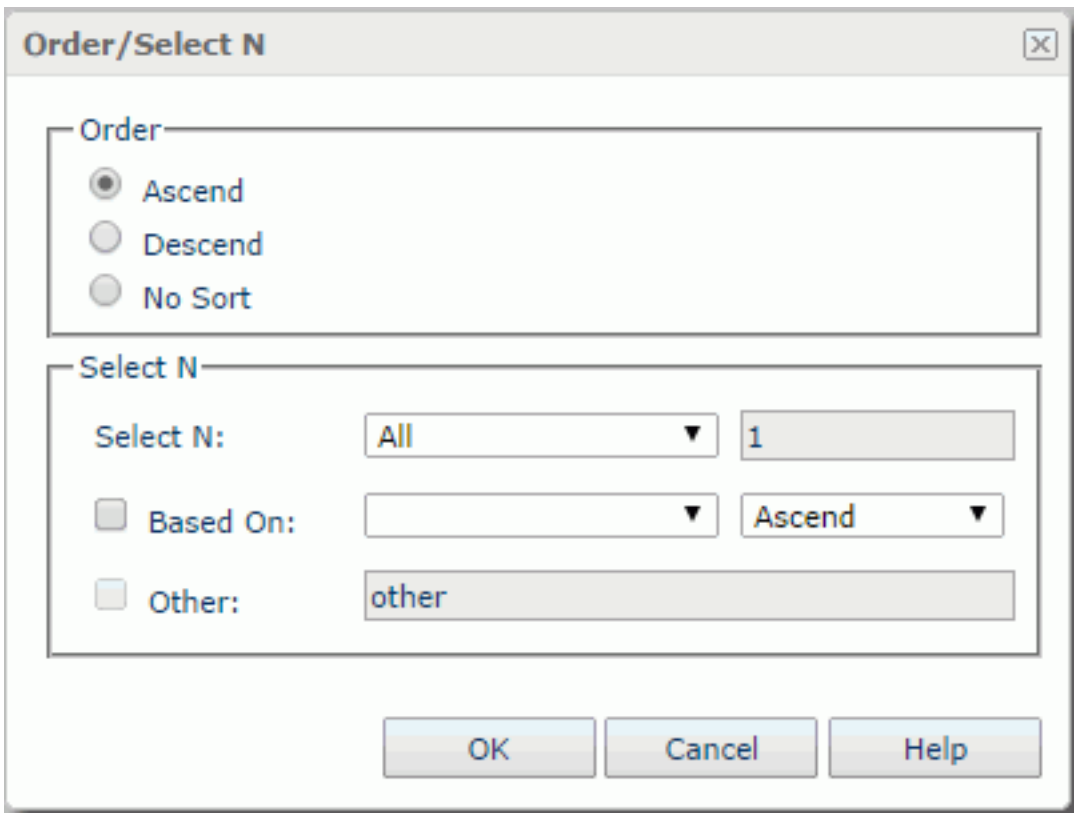
Adjusts the location of the selected tool by moving it to the right on the toolbar.

Related topics:

- Setting Page Report Studio options

Order/Select N dialog

This dialog helps you to set the sort order and the Select N condition in a chart.



Order

Specifies in which order data on the category/series axis of a chart will be displayed.

- **Ascend**
  Lists data in an ascending order.
- **Descend**
  Lists data in a descending order.
- **No Sort**
  Keeps the data in their original order.

Select N

Specifies the Select N condition to define the number of category/series values that will be displayed while others are hidden.

- **Select N**
  - **All**
    If selected, all category/series values in the chart will be displayed.
  - **Top**
    If selected, specify a number in the text box to the right and the first N category/series values in the chart will be displayed.
  - **Bottom**
    If selected, specify a number in the text box to the right and the last N category/series values in the chart will be displayed.
- **Based On**
  If checked, the category/series values will be sorted by the values of the summary selected with the direction specified. If unchecked, the category/series values will be sorted by the order specified in the Order box of the dialog.
- **Other**
  Enabled when Top or Bottom is selected from the Select N drop-down list. Check this option and then type a character string in the text box to group all the category/series values beyond the top/bottom N category/series values.

OK

Accepts the changes and closes the dialog.

Cancel

Discards the changes and closes the dialog.

Help

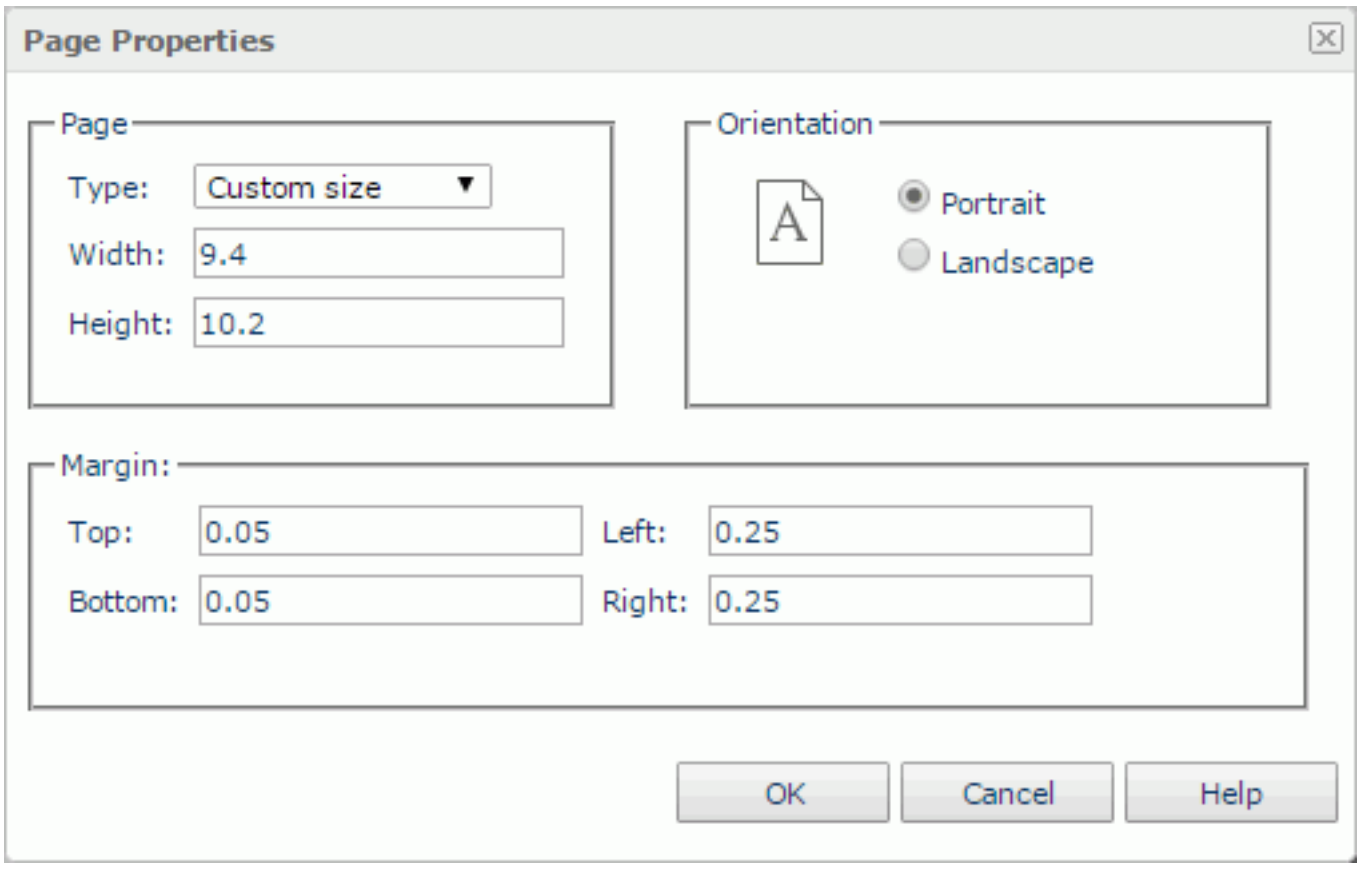
Displays the help document about this feature.

Related topics:

- Creating a chart report

Page Properties dialog

This dialog helps you to control the report page settings.



Page

Specifies the page properties.

- **Type**
  Specifies the paper type.
- **Width**
  Shows the paper width. If you select the paper type as Custom size, you can input the width of the paper here.
- **Height**
  Shows the paper height. If you select the paper type as Custom size, you can input the height of the paper here.

Orientation

Specifies the paper orientation.

- **Portrait**
  The page is positioned vertically.
- **Landscape**
  The page is positioned horizontally.

Margin

Specifies the margin properties.

- **Top**
  Specifies the distance of report data to the top edge of the page.
- **Left**
  Specifies the distance of report data to the left edge of the page.
- **Bottom**
  Specifies the distance of report data to the bottom edge of the page.
- **Right**
  Specifies the distance of report data to the right edge of the page.

OK

Applies all changes and closes this dialog.

Cancel

Does not retain any changes and exits this dialog.

Help

Displays the help document about this feature.

Parameter Control Properties dialog

This dialog helps you to edit the properties of a parameter control. It contains the following tabs:

- General tab
- Font tab
- Border tab

OK

Applies the settings and closes this dialog.

Cancel

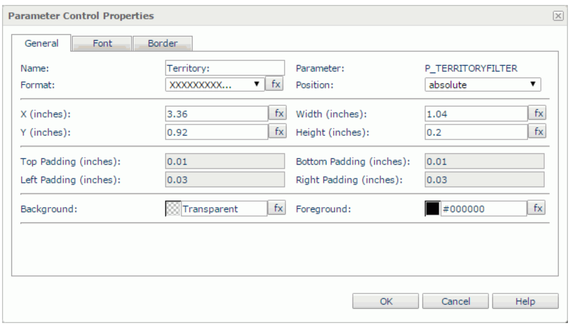
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the parameter control.



Name

Specifies the name of the parameter control.

Parameter

Displays the name of the parameter.

Format

Specifies the display format of the parameter value.

Position

Displays the position mode of the parameter control. If the parameter control is directly contained in the report body, a tabular cell, or a text box, its position mode can be modified.

- **Absolute**: The parameter control's position will be decided by its X and Y property values.
- **Static**: The parameter control will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled.

X

Specifies the X coordinate of the parameter control.

Y

Specifies the Y coordinate of the parameter control.

Width

Specifies the width of the parameter control.

Height

Specifies the height of the parameter control.

Top Padding

Specifies the space between the text of the parameter control and its top border.

Bottom Padding

Specifies the space between the text of the parameter control and its bottom border.

Left Padding

Specifies the space between the text of the parameter control and its left border.

Right Padding

Specifies the space between the text of the parameter control and its right border.

Background

Specifies the background color of the parameter control.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

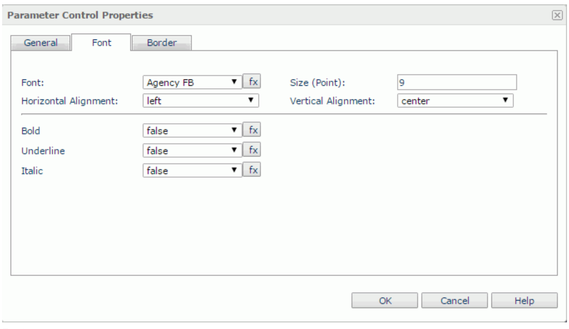
Foreground

Specifies the foreground color of the parameter control.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB.

Font tab

This tab shows the font-related information of the text in the parameter control.



Font

Specifies the font face of the text.

Size

Specifies the font size of the text.

Horizontal Alignment

Specifies the horizontal alignment mode of the text.

Vertical Alignment

Specifies the vertical alignment mode of the text.

Bold

Specifies whether to make the text bold or not.

Underline

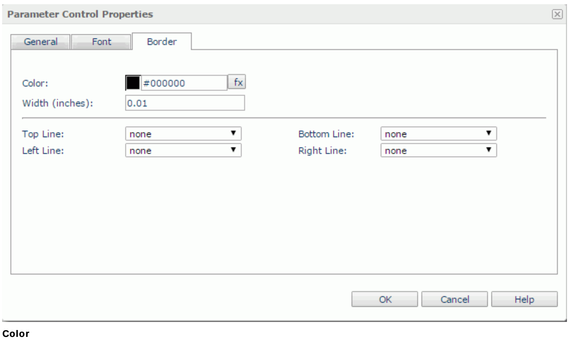
Specifies whether the text will be underlined or not.

Italic

Specifies whether to make the text italic or not.

Border tab

This tab shows information about borders of the parameter control.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Parameter Field Properties dialog

This dialog helps you to edit the properties of a parameter field. It contains the following tabs:

- General tab
- Font tab
- Border tab
- Others tab
- Display tab

OK

Applies the settings and closes this dialog.

Cancel

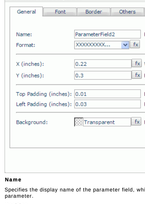
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the parameter field.



Name

Specifies the display name of the parameter field, which will be shown on the shortcut menu of the parameter.

Position

Specifies the position mode of the parameter field. If the parameter field is directly contained in the report body, a tabular cell, or a text box, its position mode can be modified.

- **Absolute**: The parameter field's position will be decided by its X and Y property values.
- **Static**: The parameter field will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled.

Format

Specifies the data format of the parameter field.

X

Specifies the X coordinate of the parameter field.

Y

Specifies the Y coordinate of the parameter field.

Width

Specifies the width of the parameter field.

Height

Specifies the height of the parameter field.

Top Padding

Specifies the space between the text of the parameter field and its top border.

Bottom Padding

Specifies the space between the text of the parameter field and its bottom border.

Left Padding

Specifies the space between the text of the parameter field and its left border.

Right Padding

Specifies the space between the text of the parameter field and its right border.

Background

Specifies the background color of the parameter field.

To change the color, click the color indicator to access the Select Color dialog, and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Foreground

Specifies the foreground color of the parameter field.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the text box.

Font tab

This tab shows the font-related information of the parameter field.

Font

Specifies the font face of the field text.

Size

Specifies the font size of the field text.

Horizontal Alignment

Specifies the horizontal alignment mode of the text in the parameter field.

Vertical Alignment

Specifies the vertical alignment mode of the text in the parameter field.

Bold

Specifies whether to make the field text bold or not.

Underline

Specifies whether the field text will be underlined or not.

Strikethrough

Specifies whether or not to attach a strikeout line to the field text.

Italic

Specifies whether to make the field text italic or not.

Autofit

Specifies whether or not to automatically adjust the width of the parameter field.

Word Wrap

Specifies whether or not to wrap the text to the parameter field width.

Ignore HTML Tag

If this option is unchecked, JReport will parse HTML tag elements in the field value while the report is to be saved as an HTML file; or the field value will appear in the HTML file the same as that in Page Report Studio (HTML tag elements in the field value, if any, will not be parsed).

Border tab

This tab shows information about borders of the parameter field.

Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Shadow

Specifies whether the borders will have a shadow effect or not.

Shadow Color

Specifies the color of the border shadow.

Others tab

You can use this tab to view and configure some miscellaneous settings.

TOC Anchor

Specifies whether or not to add the node that represents the parameter field to the TOC tree that is displayed in the TOC Browser.

Suppress When No Records

If true and no records are returned by the report, the parameter field will not be displayed.

Export to XLS

If true, the parameter field will be exported when you save the report result as an XLS file (make sure to check Data Format in the Export dialog).

Export to CSV

If true, the parameter field will be exported when you save the report result as a TXT file with Delimited Format selected.

Display tab

For a parameter field, you can change its display type to one of the following: Text, Rank, Image, Barcode, Text Field, Hidden Field, Text Area, Checkbox, Radio Button, List, Drop-down List, Image Button, Button, Submit, Reset and Hidden.

List

Specifies to display the parameter field as list.

Name

Specifies the name of the list.

Title

Specifies the title of the list.

Selected

Specifies the item to be selected.

Allow Multiple

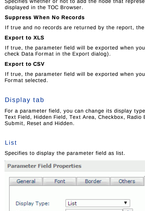
Specifies whether or not to allow multiple items to be selected.

Disabled

Specifies whether or not to make the list disabled.

Drop-down List

Specifies to display the parameter field as drop-down list.



Name

Specifies the name of the drop-down list.

Title

Specifies the title of the drop-down list.

Selected

Specifies the item to be selected.

Disabled

Specifies whether or not to make the drop-down list disabled.

Reference: For the rest display types, see Data Field Properties dialog.

Parameter Form Control Properties dialog

This dialog helps you to edit the properties of a parameter form control. It contains the following tabs:

- General tab
- Border tab
- Title tab

OK

Applies the settings and closes this dialog.

Cancel

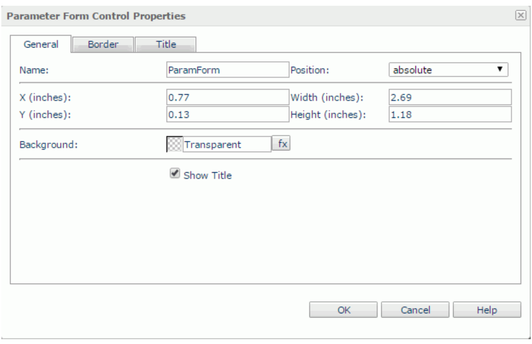
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the parameter form control.



Name

Specifies the name of the parameter form control.

Position

Displays the position mode of the parameter form control. If the parameter form control is directly contained in the report body, a tabular cell, or a text box, its position mode can be modified.

- **Absolute**: The parameter form control's position will be decided by its X and Y property values.
- **Static**: The parameter form control will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled.

X

Specifies the X coordinate of the parameter form control.

Y

Specifies the Y coordinate of the parameter form control.

Width

Specifies the width of the parameter form control.

Height

Specifies the height of the parameter form control.

Background

Specifies the background color of the parameter form control.

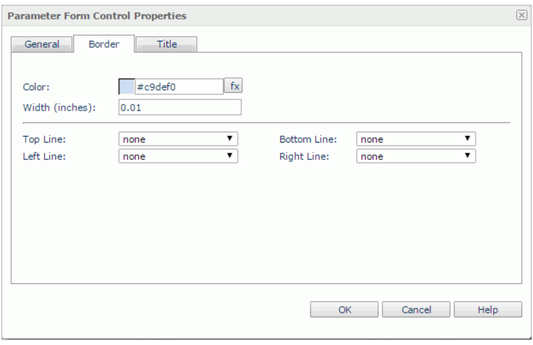
To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Show Title

Specifies whether to show the title of the parameter form control.

Border tab

This tab shows information about borders of the parameter form control.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

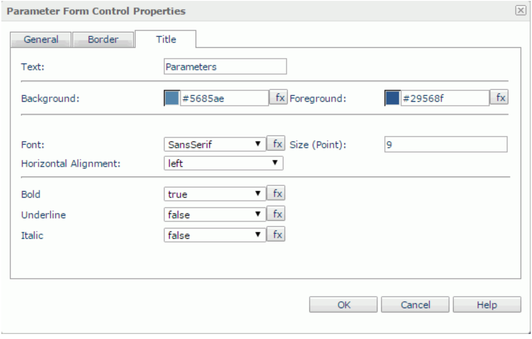
Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Title tab

This tab shows information about the title of the parameter form control. It is available when the Show Title option is selected in the General tab of the same dialog.



Text

Specifies the text of the title.

Background

Specifies the background color of the title.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Foreground

Specifies the foreground color of the title.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB.

Font

Specifies the font face of the text.

Size

Specifies the font size of the text.

Horizontal Alignment

Specifies the horizontal alignment mode of the text.

Bold

Specifies whether to make the text bold or not.

Underline

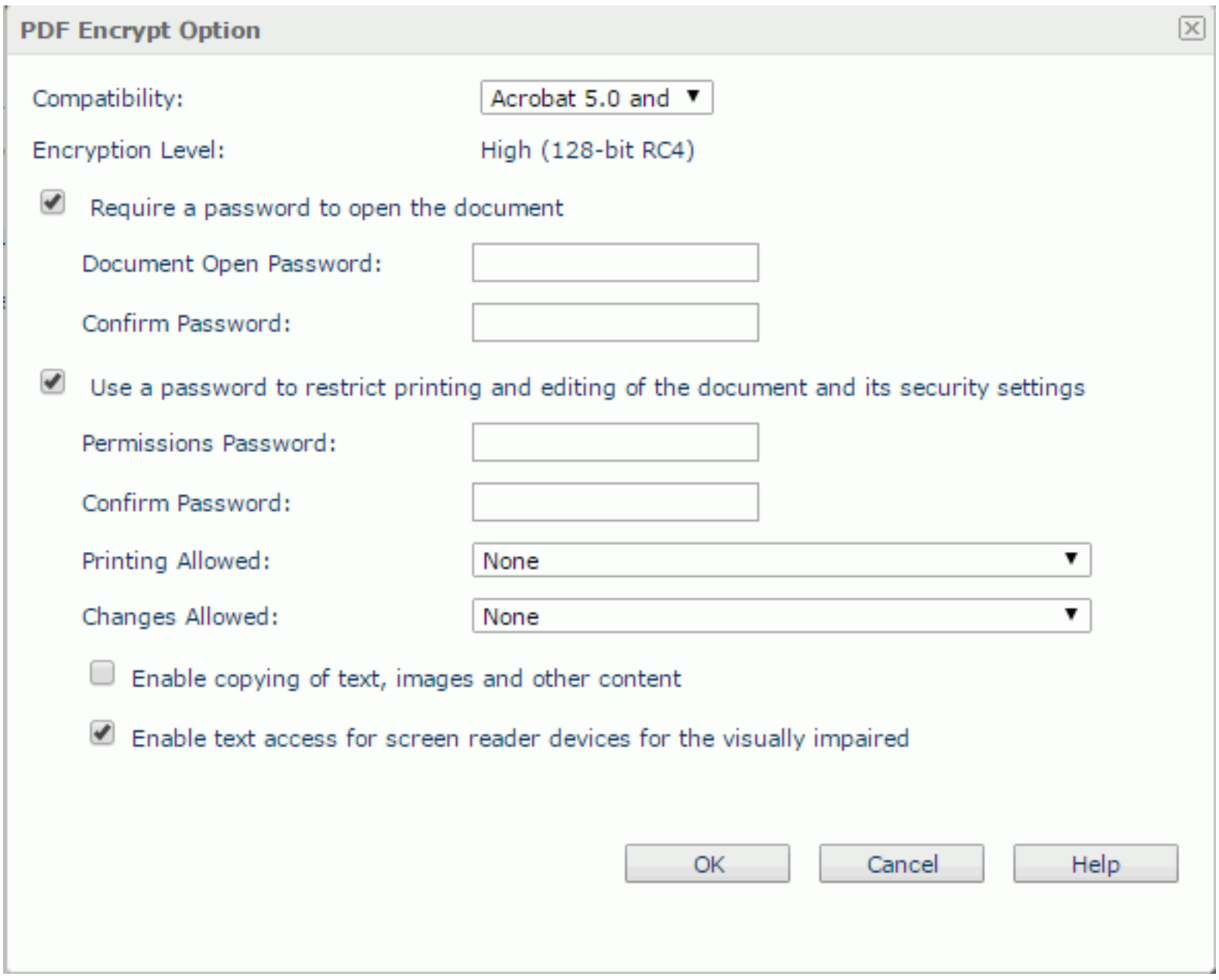
Specifies whether the text will be underlined or not.

Italic

Specifies whether to make the text italic or not.

PDF Encrypt Option dialog

This dialog helps you to set encryption options for the PDF file to be saved.



Compatibility

Specifies the encryption type to encrypt the PDF document. The option Acrobat 3.0 and later uses a low encryption level (40-bit RC4), while the other option Acrobat 5.0 and later uses a high encryption level (128-bit RC4).

Encryption Level

Shows the level of the encryption compatibility that you specified in the Compatibility drop-down list.

Require a password to open the document

Specifies the Document Open Password to prevent others from opening the document without authorization.

- **Document Open Password**
  Specifies the password to prevent others from opening the document without authorization.
- **Confirm Password**
  Confirms the password you have specified in the Document Open Password text box.

Use a password to restrict printing and editing of the document and its security settings

Specifies the Permission Password to prevent others from printing and editing the document. The password you specify here cannot be the same as the one that you use to open the document.

- **Permissions Password**
  Specifies the password to prevent others from printing and editing.
- **Confirm Password**
  Confirms the password you have specified in the Permissions Password text box.
- **Printing Allowed**
  Specifies the printing quality for the PDF document.
- **Changes Allowed**
  Specifies the editing actions that are permitted in the PDF document.

Enable copying of text, images and other content

Allows others to select and copy the contents of the PDF document.

Enable text access for screen reader devices for the visually impaired

Allows visually impaired users to read the PDF document with window readers. This option is available only if the Compatibility option is set to Acrobat 5.0 or later Acrobat version.

OK

Saves the settings and exits the dialog.

Cancel

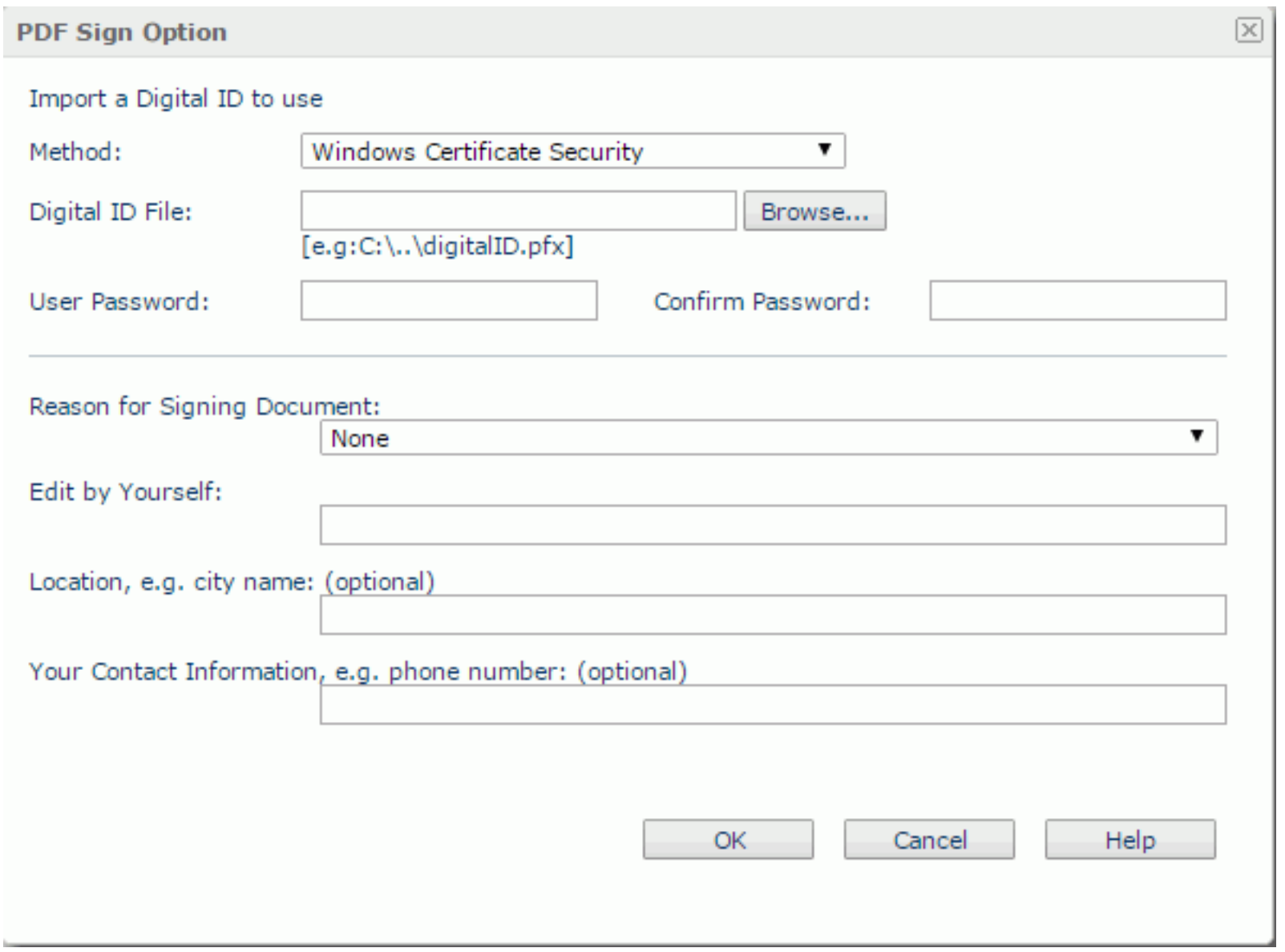
Cancels the settings and closes the dialog.

Help

Displays the help document about this feature.

PDF Sign Option dialog

This dialog helps you to set signing options for the PDF file to be saved.



Method

Specifies the digital signing method.

Digital ID File

Specifies whether to use a digital ID file from a specified directory.

User Password

Specifies a password for this digital ID file.

Confirm Password

Confirms the password.

Reason for Signing Document

Specifies the reason for signing the document.

Edit by Yourself

Edits the reason for signing the document. It is enabled when I agree to 'specified' portions of this document is selected from the Reason for Signing Document drop-down list.

Location

Specifies your location accordingly.

Your Contact Information

Specifies your contact information accordingly.

OK

Applies the settings and closes this dialog.

Cancel

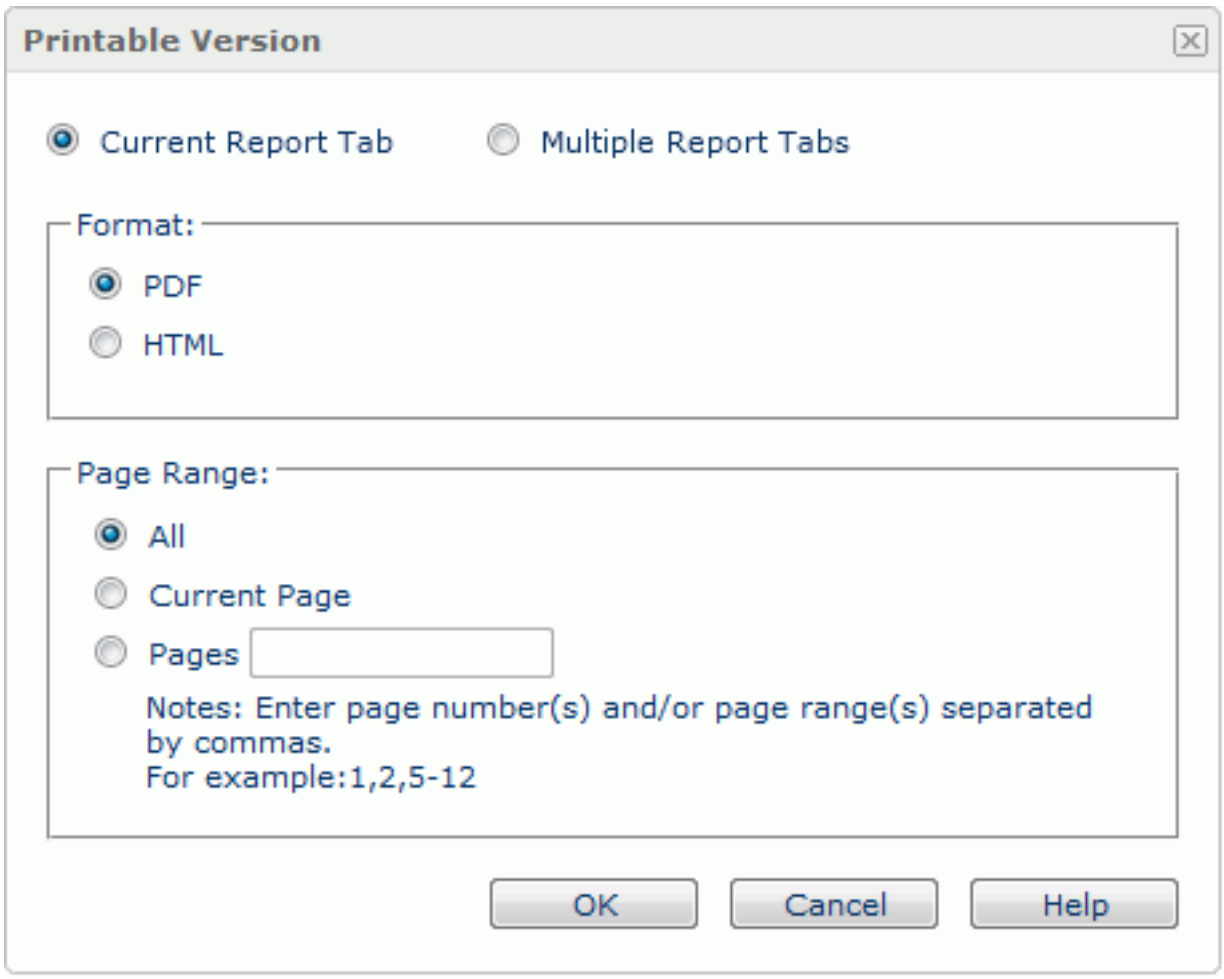
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

Printable Version dialog

This dialog helps you to set the print settings.



Prints the current report tab of the page report.

Multiple Report Tabs

Prints multiple report tabs of the page report.

- **Select Report Tabs**
  Specifies the report tabs which you want to print. The selected report tabs will be printed in the specified format in the list order. If the report has only one report tab, it is selected by default.
- ⇧
  Moves the specified report tab one step up.
- ⇩
  Moves the specified report tab one step down.

Format

Specifies the format in which the report result will be printed.

- **PDF**
  Specifies to print the report result to a PDF file.
- **HTML**
  Specifies to print the report result to an HTML file.

Page Range

Specifies the range of the pages that are to be printed.

- **All**
  Specifies that all pages will be printed.
- **Current Page**
  Specifies that only the current page will be printed.
- **Pages**
  Specifies page numbers and/or page ranges separated by commas.

OK

Prints the report result with the specified settings to a PDF or HTML file and opens it in an associated program with which you can print the result to a printer.

Cancel

Cancels the print settings and closes this dialog.

Help

Displays the help document about this feature.

Query Filter dialog

This dialog helps you to apply a filter to the business view used by the specified data component to narrow down its data scope. It has the following two modes:

- Basic
- Advanced

Advanced / Basic

Switches the dialog to the advanced/basic mode.

OK

Applies the settings and closes the dialog.

Cancel

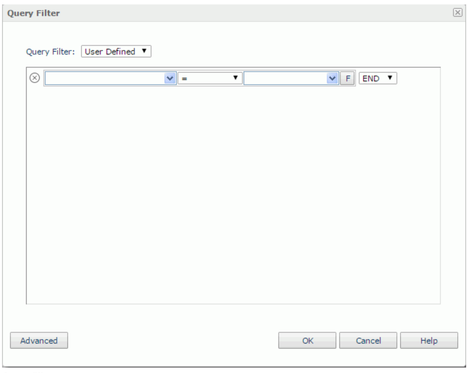
Closes the dialog and discards any changes.

Help

Displays the help document about this feature.

Basic mode

The basic mode provides function for creating simple filter conditions which are connected by the AND and OR operators.



Query Filter

Specifies the filter you want to apply to the business view.

- **User Defined**
  Specifies to create a user defined filter to apply to the business view.
- **Predefined filters**
  The names of the predefined filters which were created on the business view in JReport Designer. You can choose one of them to apply.
  - **Edit**
    Edits the selected predefined filter. Once a predefined filter is edited, it will be saved as a user defined filter in the business view.
  - **Description**
    Displays the description of the selected predefined filter.

Deletes the current condition line.

Field

Specifies the field to be filtered.

Operator

Specifies the operator to compose the filter expression.

- **=**
  Equal to
- **>**
  Greater than
- **>=**
  Greater than or equal to
- **<**
  Less than
- **<=**
  Less than or equal to
- **!=**
  Not equal to
- **[not] in**
  Causes an enumerated list of values to appear in the WHERE clause predicate, used for evaluating for a true condition.
- **[not] like**
  Like string pattern matching operator is used to compare the first expression string value to the pattern string (the second expression). If you want to use wildcard character in the pattern string, only "_" and "%" are supported.
- **[not] between**
  Allows the system to evaluate whether or not data values are located between a range of values indicated in the predicate.
- **is [not] null**
  It is used in WHERE clause predicates to match null values occurring in a specified data field.

Value

Specifies the value of how to filter the field. You can either enter the value manually in the text box or select a value from the drop-down list. Click the button F or V to switch to filter the field by field or value. When you type in the value manually, if multiple values are required, they should be separated with ",", and if "," or "\" is contained in the values, write it as "\," or "\\".

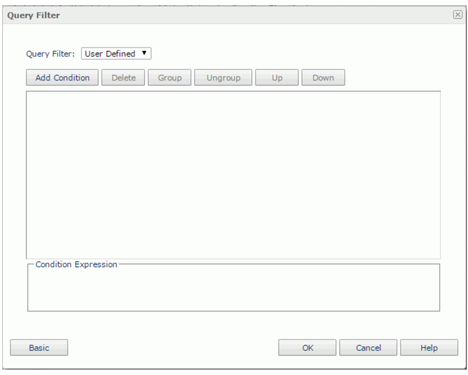
Logic

Lists the logic operator.

- **AND**
  Logic operator AND which is applied to this and the next line.
- **OR**
  Logic operator OR which is applied to this and the next line.

Advanced mode

The advanced mode enables the building of more complex filter conditions via the grouping of conditions.



Query Filter

Specifies the filter you want to apply to the business view.

- **User Defined**
  Specifies to create a user defined filter to apply to the business view.
- **Predefined filters**
  The names of the predefined filters which were created on the business view in JReport Designer. You can choose one of them to apply.
  - **Edit**
    Edits the selected predefined filter. Once a predefined filter is edited, it will be saved as a user defined filter in the business view.
  - **Description**
    Displays the description of the selected predefined filter.

Add Condition

Adds a new condition line.

Delete

Deletes the selected condition line.

Group

Makes the selected conditions in one group. Conditions can also be added to an existing group by selecting the conditions and the group while holding the Ctrl button, and then clicking the Group button.

Ungroup

Makes the selected condition ungrouped.

Up

Moves the selected condition or group up to a higher level.

Down

Moves the selected condition or group down to a lower level.

Logic

Specifies the logic operator.

- **AND**
  Logic operator AND which is applied to this and the next line.
- **OR**
  Logic operator OR which is applied to this and the next line.
- **AND NOT**
  Logic operator AND NOT which is applied to this and the next line.
- **OR NOT**
  Logic operator OR NOT Not which is applied to this and the next line.

Field

Specifies the field to be filtered.

Operator

Specifies the operator to compose the filter expression.

Value

Specifies the value of how to filter the field. You can either enter the value manually in the text box or select a value from the drop-down list. Click the button F or V to switch to filter the field by field or value. When you type in the value manually, if multiple values are required, they should be separated with ",", and if "," or "\" is contained in the values, write it as "\," or "\\".

Condition Expression

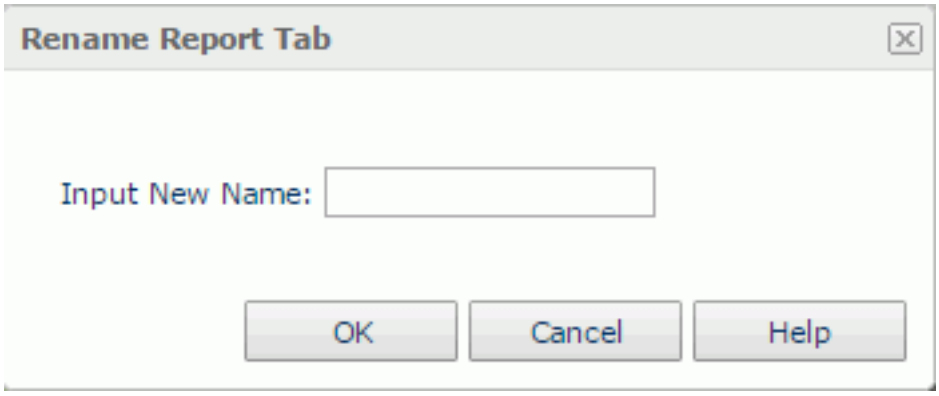
Displays the SQL statement of the filter condition.

Related topics:

- Applying filters to business views

Rename Report Tab dialog

This dialog helps you to rename the current report tab.



Input New Name

Specifies the new name for the report tab.

OK

Applies the new name to the report tab and closes this dialog.

Cancel

Discards the setting and closes this dialog.

Help

Displays the help document about this feature.

Report Body Properties dialog

This dialog helps you to set the properties of the report body. It contains the following tabs:

- General tab
- Border tab
- Others tab

OK

Applies the settings and closes this dialog.

Cancel

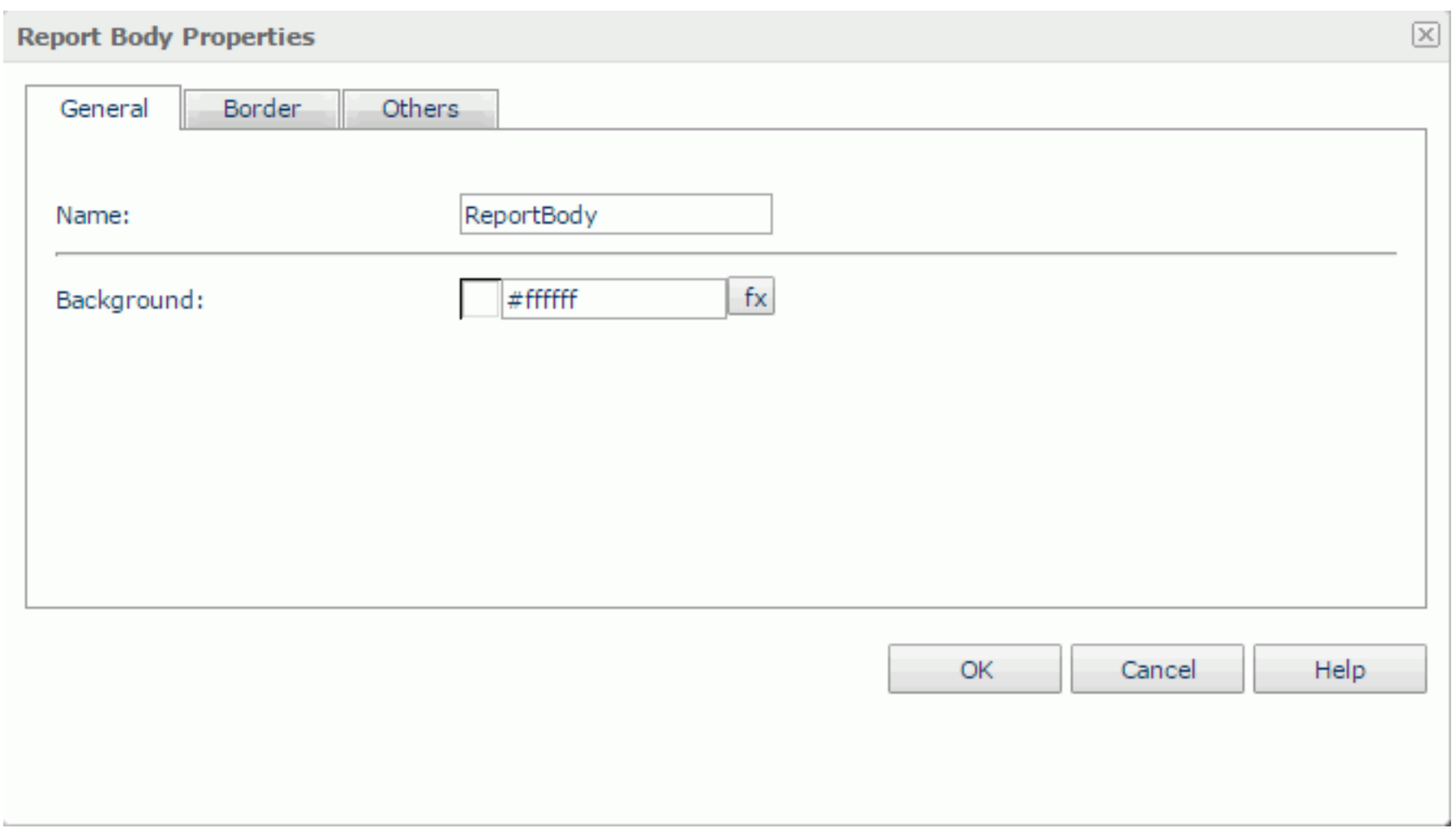
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the report body.



Name

Specifies the display name of the report body, which will be shown on its shortcut menu.

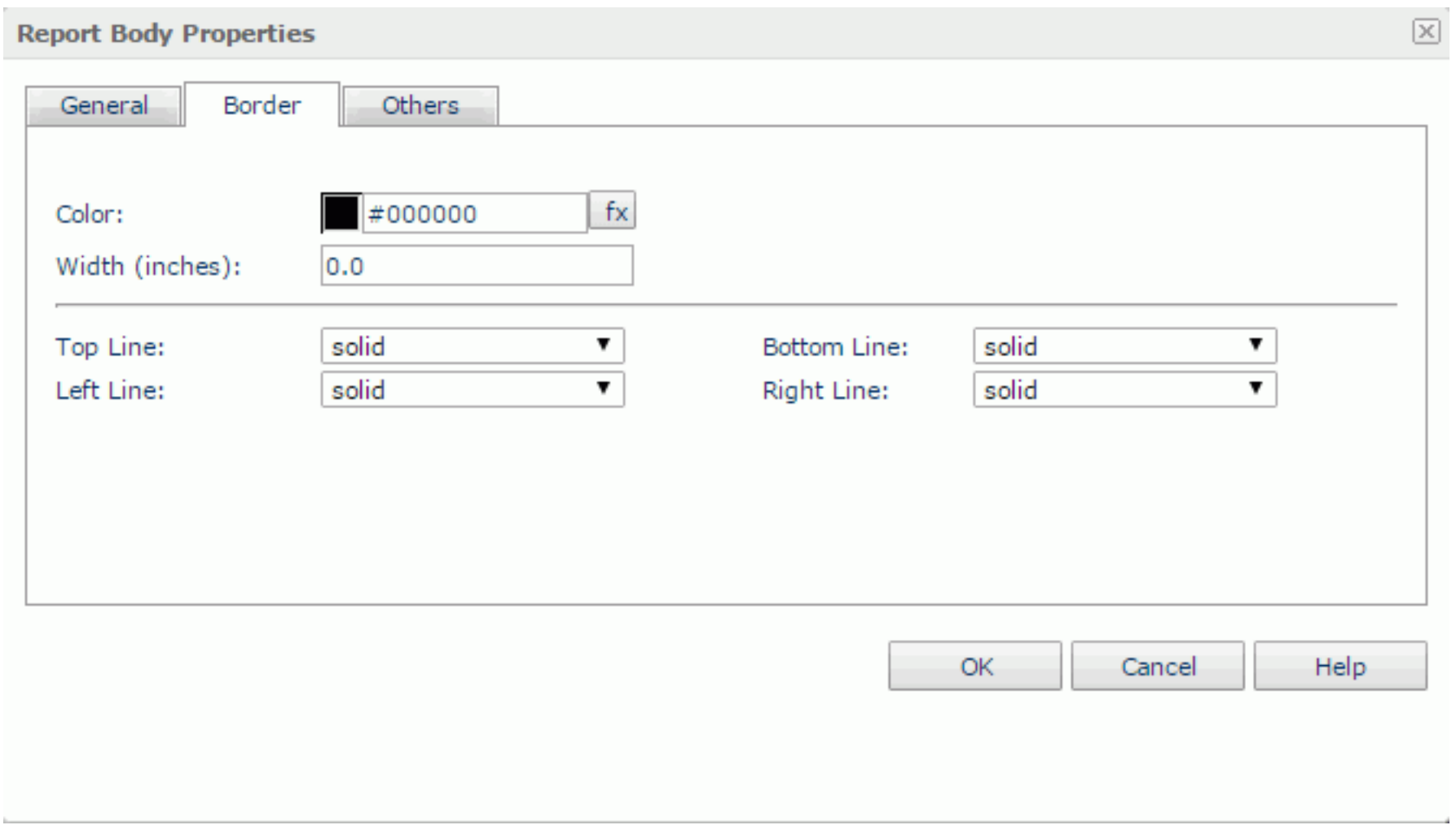
Background

Specifies the background color and fill effect of the report body.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the report body.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

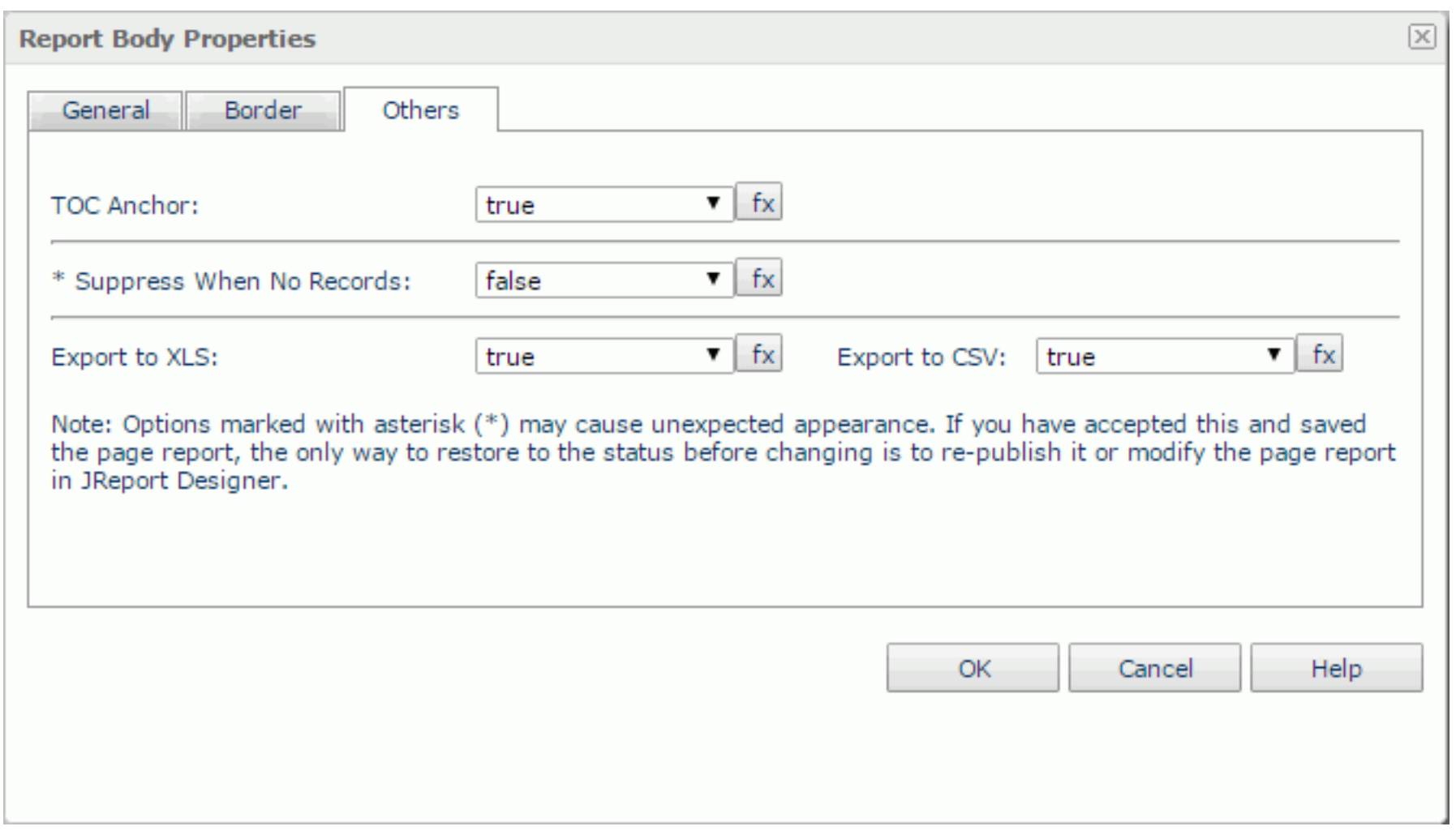
Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Others tab

You can use this tab to view and configure some miscellaneous settings.



TOC Anchor

Specifies whether or not to add the node that represents the report body to the TOC tree that is displayed in the TOC Browser.

Suppress When No Records

If true and no records are returned by the report, the report body will not be displayed.

Export to XLS

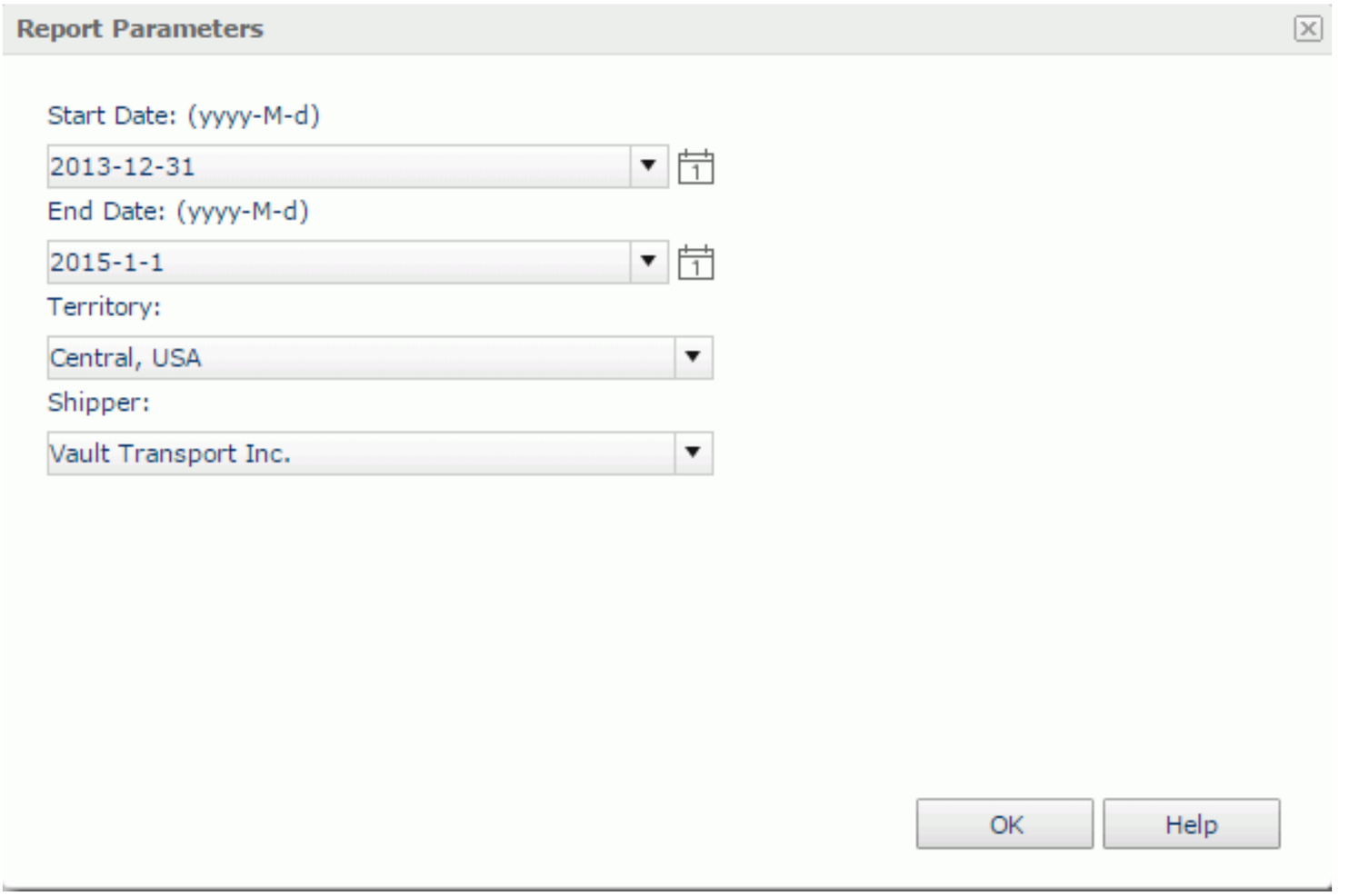
If true, the report body will be exported when you save the report result as an XLS file (make sure to check Data Format in the Export dialog).

Export to CSV

If true, the report body will be exported when you save the report result as a TXT file with Delimited Format selected.

Report Parameters dialog

This dialog appears when you click Finish in the report wizard to run a page report with parameters or click Menu > Report > Change Parameters if the current report has parameters. It helps you to input parameter values for a report.



The way to specify a parameter value varies with the type and properties of the parameter. Here are several ways you can use to specify parameter values:

- In the parameter value combo box, select the required one from the drop-down list or input the value manually.
- Select or unselect the checkbox to specify a Yes/No value.
- Click the button ... to specify multiple values in the Enter Values dialog.
- Click the calendar button to specify a date and time value using either calendar or expression in the Calendar dialog. If you use an expression to specify the value, when you hover the mouse pointer over this value, a tip will appear showing its expression. Click on the value, the expression will be displayed in the value text box and you can edit the expression in the text box directly if you want. After editing, when you click elsewhere outside of the value text box:
  - If the edited expression is correct, a new value calculated by the expression will be displayed in the parameter value text box.
  - If the edited expression is wrong, no value will be created and the expression itself will be displayed and highlighted in red in the value text box. Correct the expression, or click the calendar button to specify a value.

OK

Accepts the parameter values for further processing.

Help

Displays the help document about this feature.

Report Tab Properties dialog

This dialog helps you to set the properties of a report tab. It contains the following tabs:

- General tab
- Others tab

OK

Applies the settings and closes this dialog.

Cancel

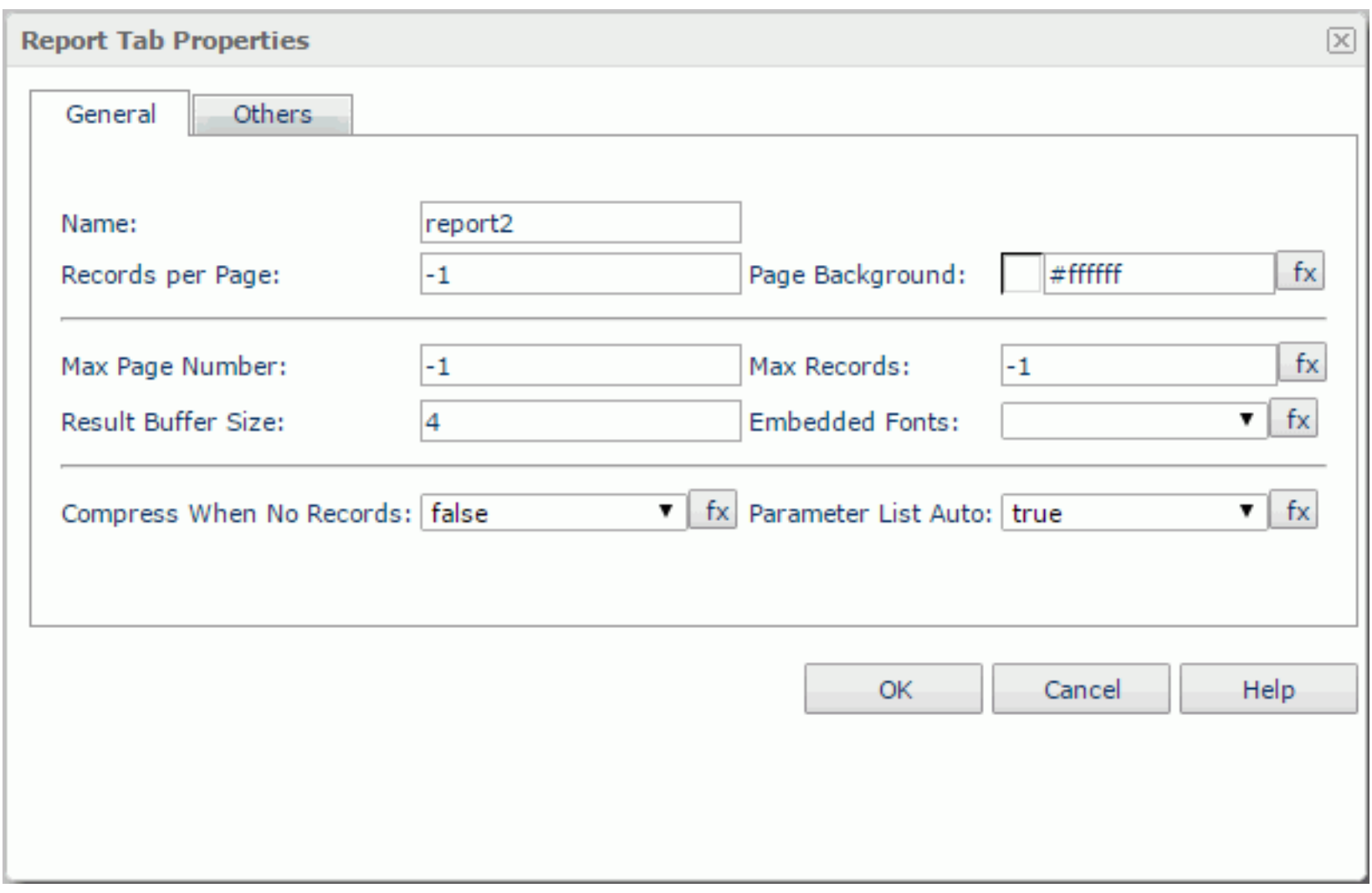
Discards the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the report tab.



Name

Specifies the report tab name.

Records per Page

Specifies the number of records in each page for the data buffer.

Page Background

Specifies the page background color.

Max Page Number

Specifies the maximum number of pages in the data buffer.

Max Records

Specifies the maximum number of records you want to display for the whole report tab.

Result Buffer Size

Specifies the size of the result buffer for storing the report result.

Embedded Fonts

Specifies the True Type Fonts that have been used in the report tab. This property is only for exporting PDF files.

Compress If No Record

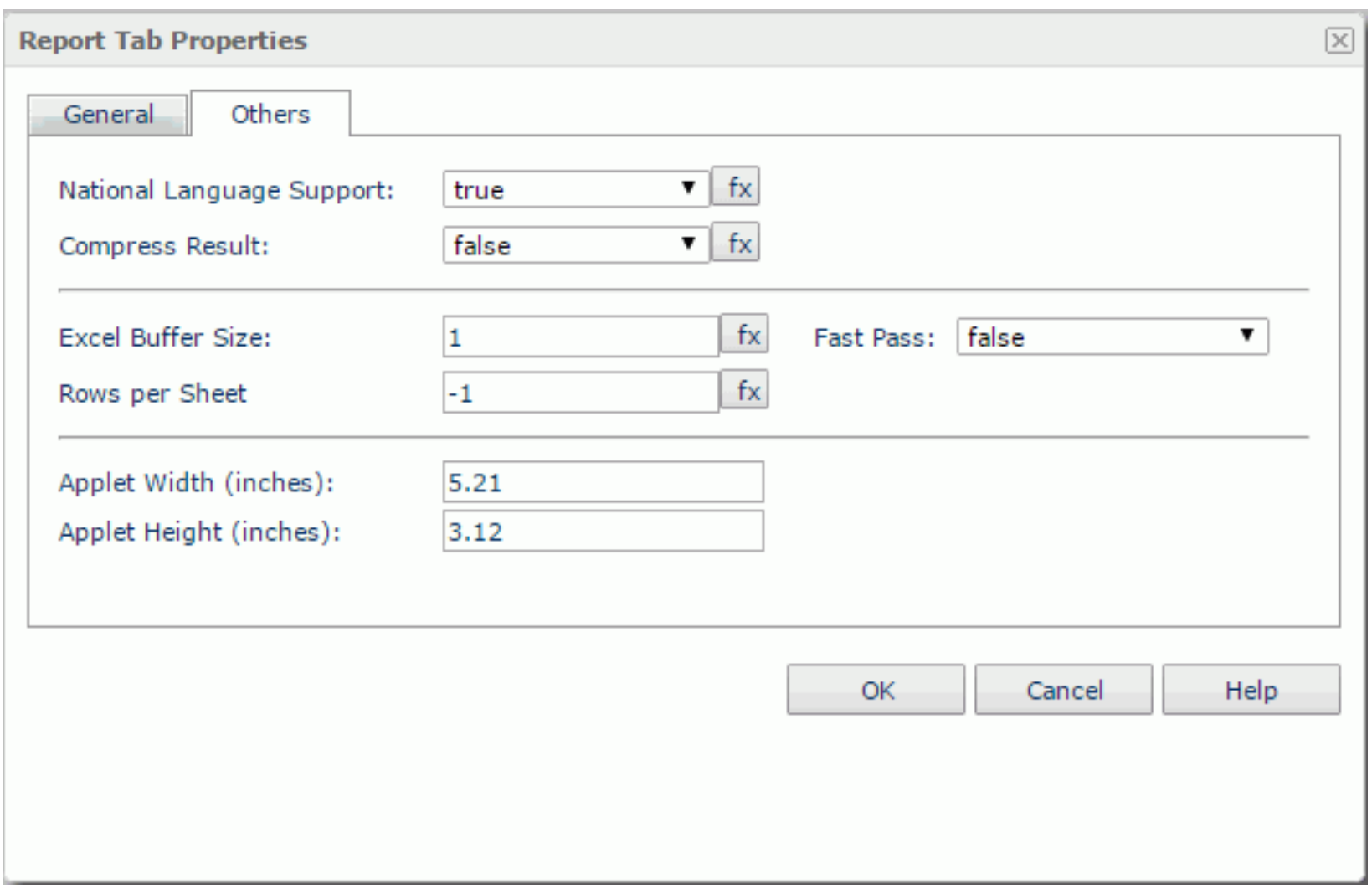
If this option is set to true and there is no record retrieved to the report tab, the report page will be compressed.

Parameter List Auto

If true, values defined in the catalog will be used as the default parameter values; otherwise, JReport will get the default values from a specified class file.

Others tab

This tab shows some miscellaneous information of the report tab.



National Language Support

Specifies a language profile file.

Compress Result

Specifies whether or not to compress the exported RST file.

Excel Buffer Size

Specifies the size of the Excel buffer to store the XLS format report result.

Fast Pass

If true, the performance of the engine when saving the report result to a CSV format file on JReport Server will be improved.

Rows per Sheet

Specifies the maximal number of rows for every worksheet when exporting the result to an XLS file.

Applet Width

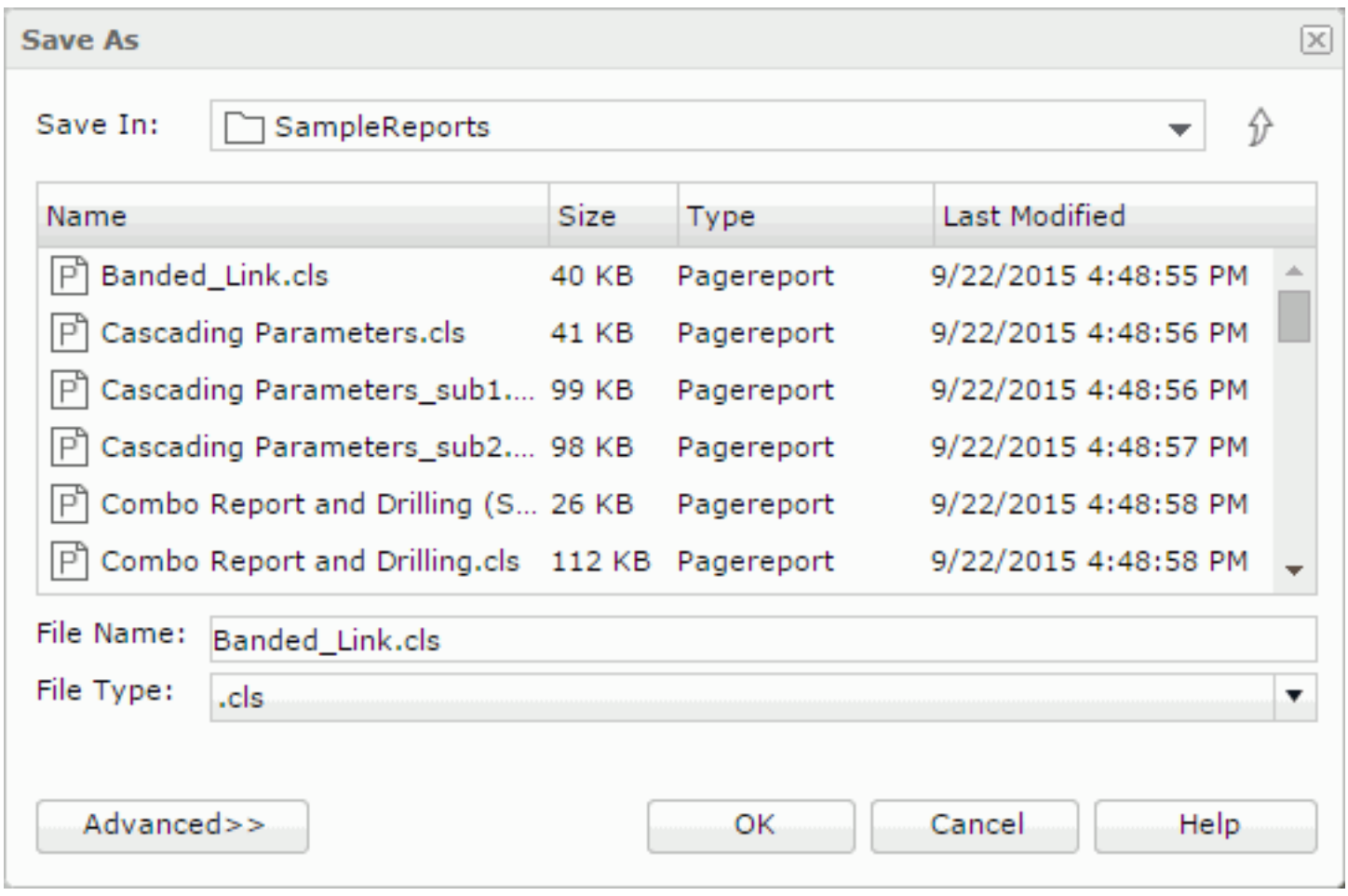
Specifies the applet width for the exported applets to be run in a web server.

Applet Height

Specifies the applet height for the exported applets to be run in a web server.

Save As dialog

This dialog helps you to save a report as a new file.



Save In

Specifies the directory in the server resource tree where you want to save the report. The folder may be Public Reports or My Reports. Use the button to go to the parent folder.

The resource table shows the resources in the current directory. Click the column names to change the order of the report in the table list if required.

- **Name**
  Displays the file names.
- **Size**
  Displays the file size.
- **Type**
  Displays the file type.
- **Last Modified**
  Displays the last modified time of the file.

File Name

Specifies the name for the saved report, without suffix.

File Type

Specifies the type of the saved report.

Advanced/Simple

Displays the following settings or hides them.

- **Catalog**
  Shows the catalog that the report uses.

  The following two options are available only when Select Catalog Linking Model is checked in the Profile > Customize Profile > Page Report Studio > Properties > Advanced tab.

  - **Set Original Catalog as Linked Catalog into Saved Report**
    If checked, the saved report will be linked with the catalog and the saved report will run with the catalog no matter whether the two are in the same directory. If later the catalog is updated, the saved report will run with the latest version of the catalog.
  - **Set Catalog Copy to Target Folder**
    If checked, the catalog will be copied to the directory where the report is saved and the saved report will run with the copied catalog.
- **Save Sort Criteria**
  Specifies whether or not to save the changes made by sorting.
- **Save Filter Criteria**
  Specifies whether or not to save the changes made by filtering.
- **Description**
  Provides a description for the saved report.

OK

Saves the report with the specified settings and closes this dialog.

Cancel

Does not save the report as a new file and closes the dialog.

Help

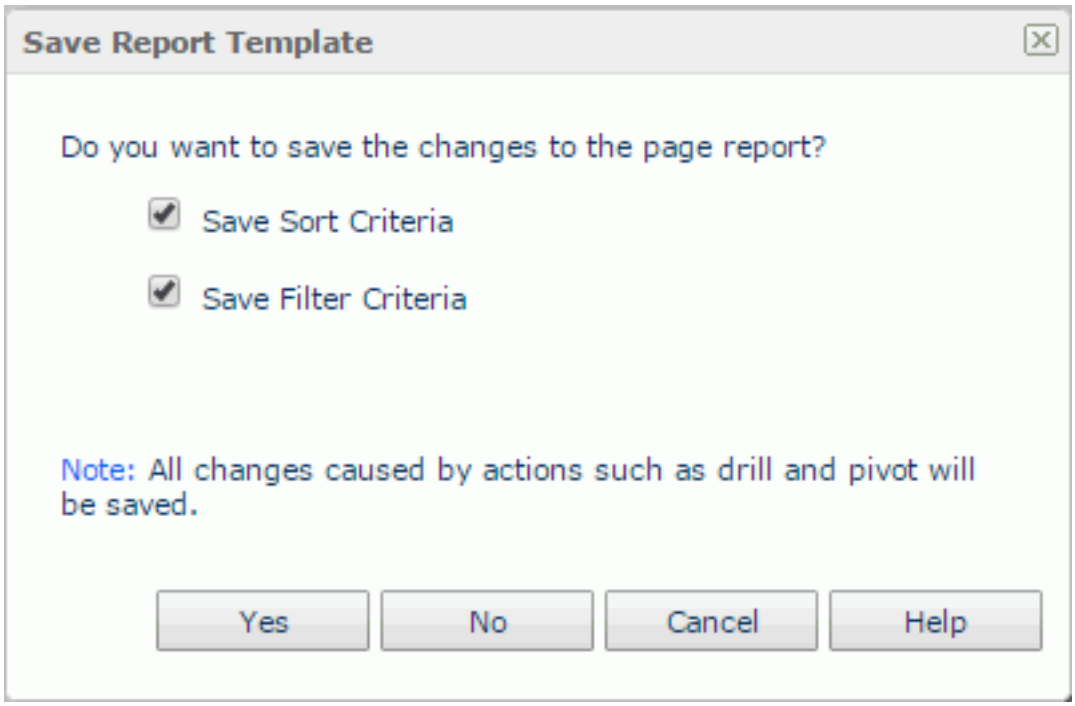
Displays the help document about this feature.

Related topics:

- Saving the report

Save Report Template dialog

This dialog requires your confirmation on whether or not to save the report template when saving a report. You can use the option Pop Up Save Criteria Dialog in the Profile > Customize Profile > Page Report Studio > Properties > Default tab to control whether or not to display this dialog when saving a report.



Save Sort Criteria

Specifies whether or not to save the sorting criteria with the report.

Save Filter Criteria

Specifies whether or not to save the filtering criteria with the report.

Yes

Applies the settings and closes this report.

Cancel

Does not save the changes to the report and returns to the report.

Help

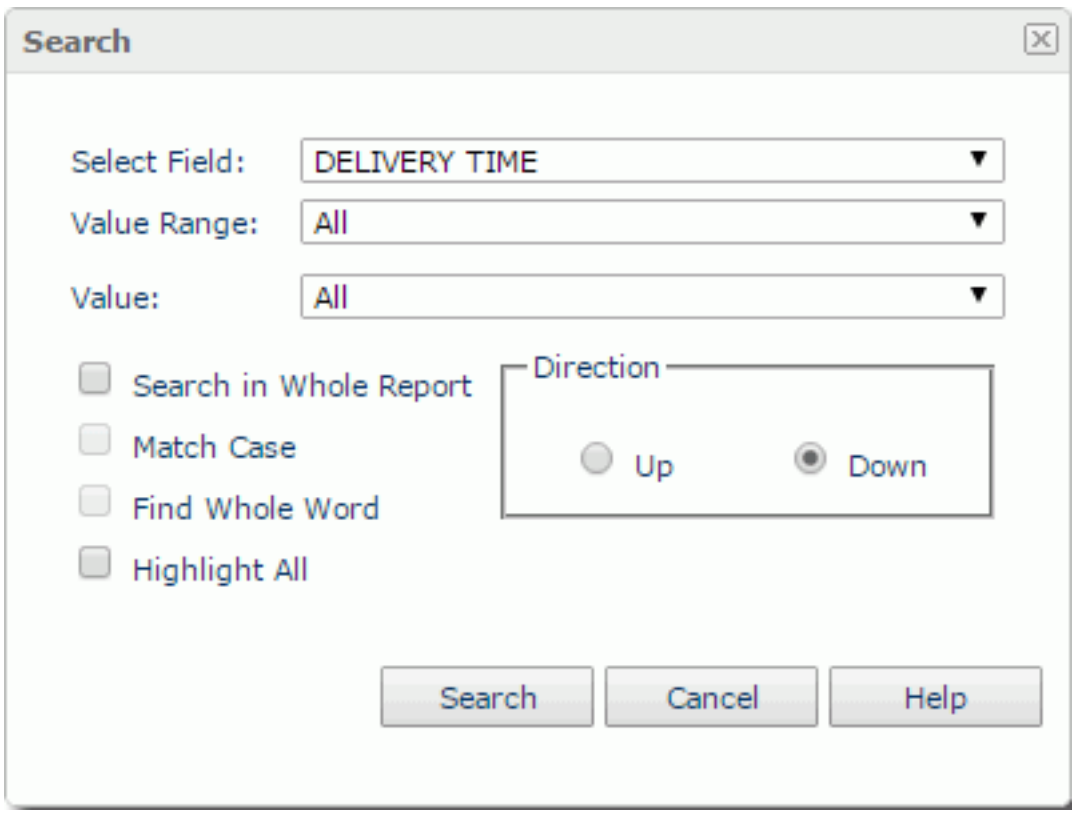
Displays the help document about this feature.

Related topics:

- Saving the report

Search dialog

This dialog helps you to find specific text in a report.



The content can be found in two places - in the values of a certain field, or in the report content.

- To find a certain field value, select the field from the Select Field drop-down list, define the value range in the Value Range drop-down list and then select the value from the Value drop-down list.
- To find text in the report content, check the **Search in Whole Report** checkbox, type the search content in the Value box.

Select Field

Specifies the field in which you want to find the text. Disabled when Search in Whole Report is checked.

Value Range

Specifies the range of the displayed values so that you can select a required value quickly from the Value field. Disabled when Search in Whole Report is checked.

Note: If All is selected in the Value Range drop-down list, the only item in the Value drop-down list will be All and you cannot change the value, in which case, when you submit the search, JReport will search for all the values of the selected field.

Value

Specifies the text you want to find. When Search in Whole Report is unchecked, you can select a value from the drop-down list.

Search in Whole Report

Finds text in the report content. The Select Field drop-down list and the Value Range drop-down list are disabled when this box is checked.

Match Case

Finds text only if it matches the capitalization of the text you have entered.

Find Whole Word

Finds text only if it matches a whole word.

Highlight All

Highlights all the matching text.

Direction

Specifies the searching direction.

- **Up**
  Searches from the last found string to the beginning of the report.
- **Down**
  Searches from the last found string to the end of the report. This option is selected by default.

Search

Searches the report or field value for the next match of the specified text.

Cancel

Cancels the operation and closes this dialog.

Help

Displays the help document about this feature.

Related topics:

- Searching for text in a report

Select Color dialog

This dialog helps you to specify a color. It contains the following tabs:

- Swatches tab
- Color Picker tab

OK

Closes this dialog and returns the color value specified in the current tab.

Cancel

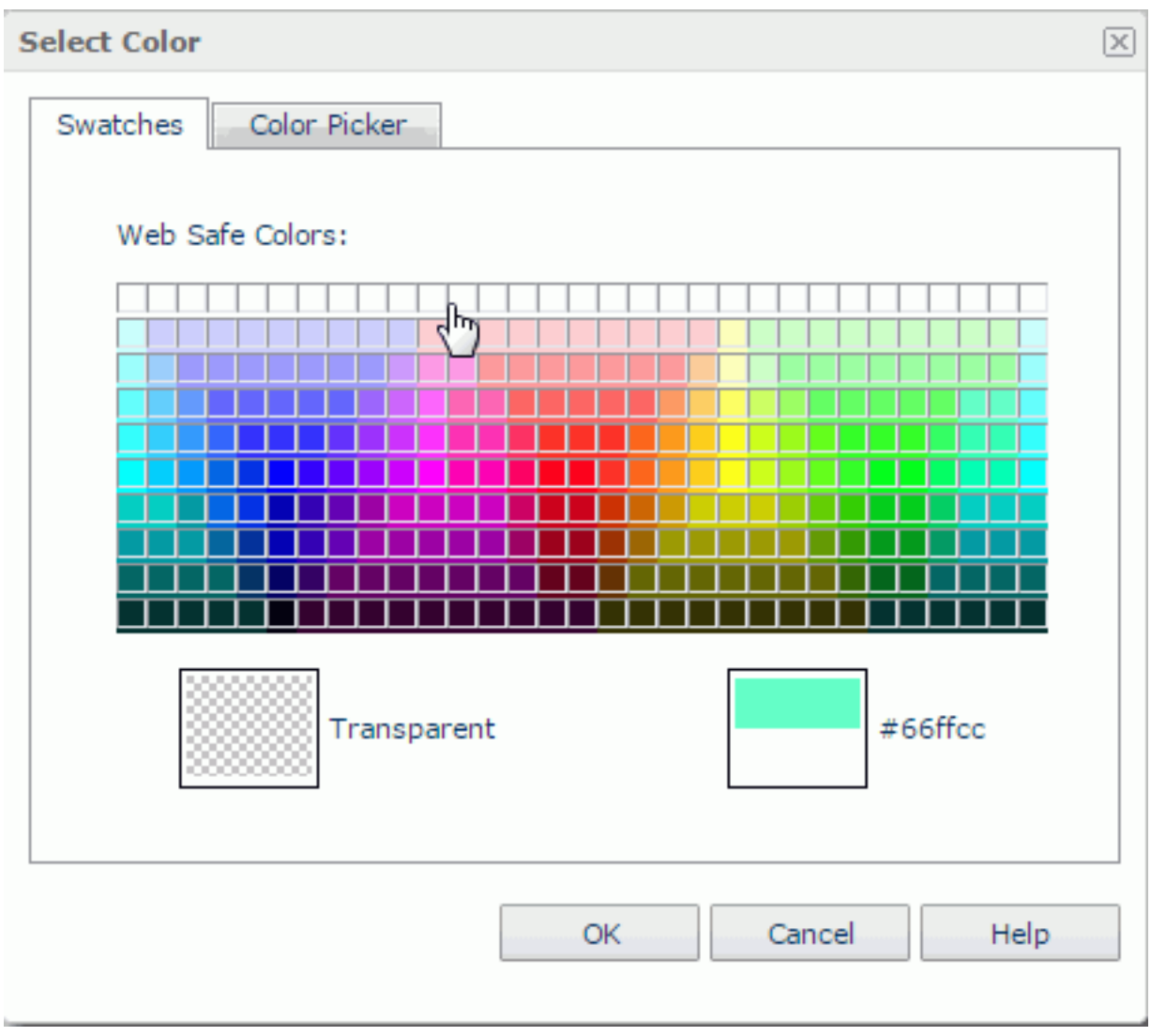
Closes this dialog without changing the color value.

Help

Displays the help document about this feature.

Swatches tab

This tab allows you to select one from the web safe color swatches.



Web Safe Colors

Lists the web safe colors for you to select one.

Transparent

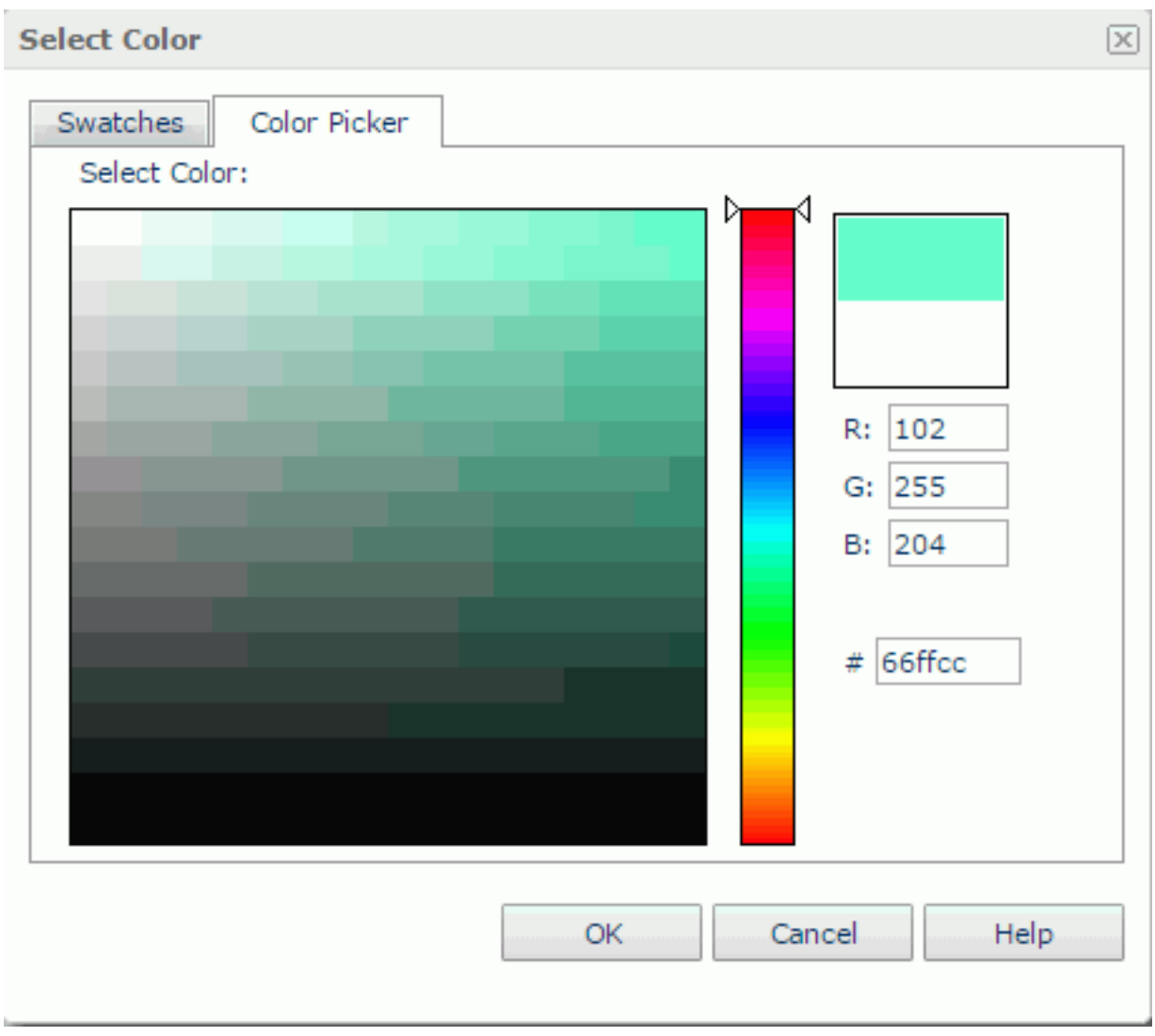
Click this swatch to delete the color and specify the object to be transparent.

Color indicator

The upper part will be displayed in the color you selected from Web Safe Colors and labeled the hexadecimal value of the color (or "Transparent" if you click the Transparent swatch). The lower part is always displayed in the original color. Clicking the lower part will restore the color to the original.

Color Picker tab

This tab allows you to customize the color in Page Report Studio.



Select Color

Displays a color matrix. To define a color, click anywhere in the matrix. Change saturation by moving the pointer horizontally, and change brightness by moving the pointer vertically.

Color slider

Drag the slider to change the hue.

Color indicator

The upper part will be displayed in the color you have newly defined. The lower part is always displayed in the original color. Clicking the lower part will restore the color to the original.

R

Specifies the amount of red in a color.

G

Specifies the amount of green in a color.

B

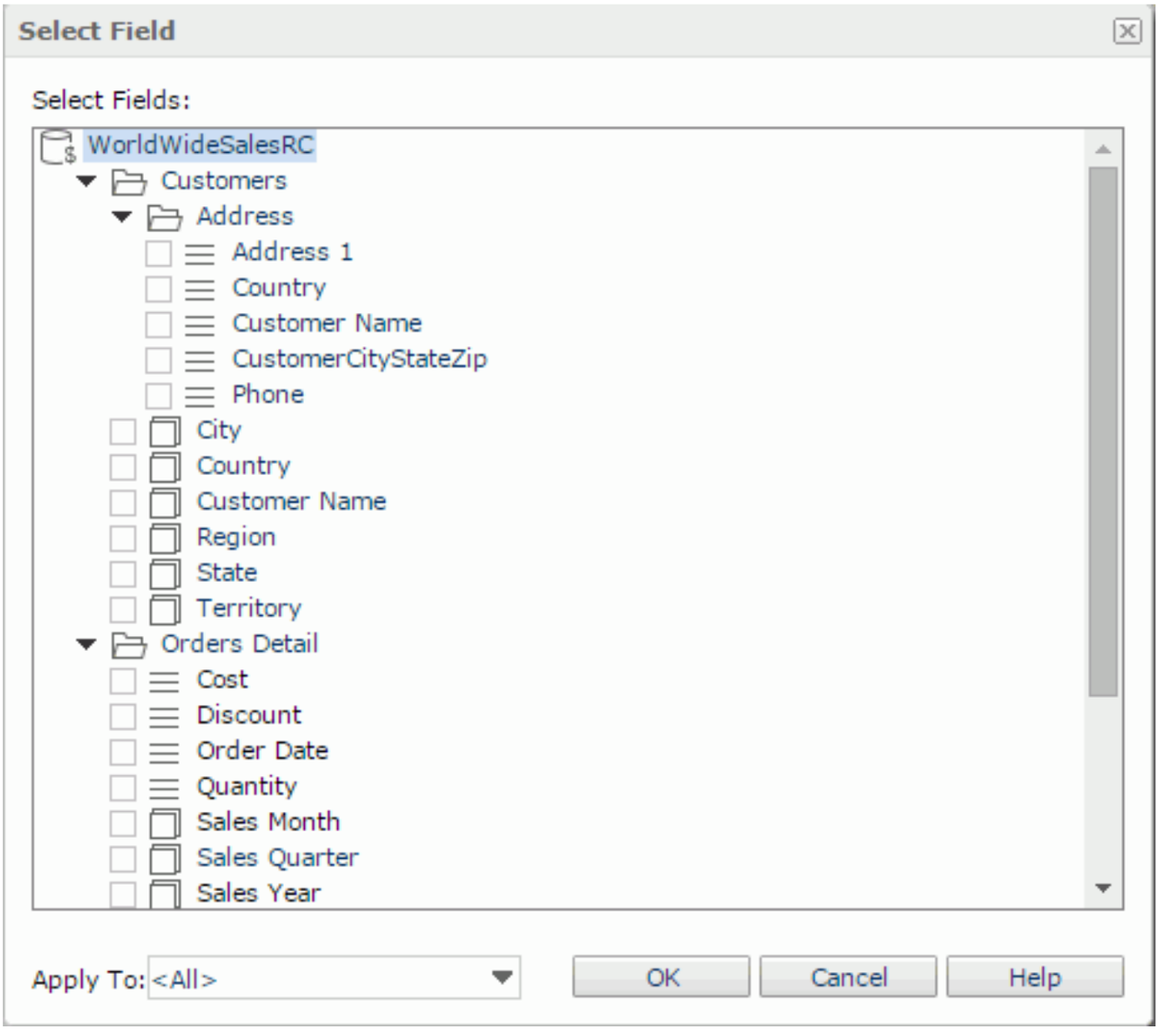
Specifies the amount of blue in a color.

Color value

Specifies the hexadecimal value of the color.

Select Field dialog

This dialog helps you to select some fields of the same type to replace the current fields in a filter control.



Select Fields

Specifies the fields which will replace the current fields in the filter control. The selected fields should be of the same type. All uncomparable data type fields cannot be bound to filter controls, such as Binary, Blob, Clob, Longvarchar, Longvarbinary, and Varbinary.

Apply To

Specifies the components to which the filter created with the filter control will be applied.

OK

Applies the selected field to the filter control and closes the dialog.

Cancel

Cancels the selection and closes the dialog.

Help

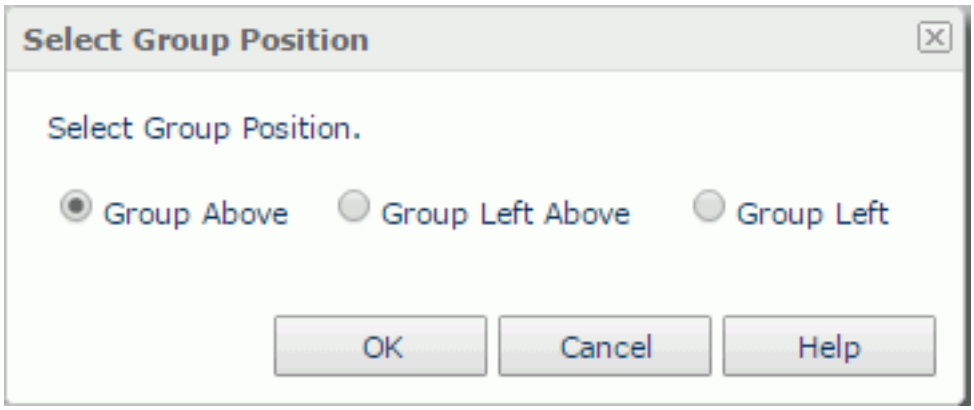
Displays the help document about this feature.

Related topics:

- Using filter control to filter report data

Select Group Position dialog

The dialog helps you to specify the position of the group by field when you convert a table detail column to a group column.



Group Above

Specifies to place the group by field in its own row above the detail columns.

Group Left Above

Specifies to place the group by field in its own row and column above and left of the detail columns.

Group Left

Specifies to place the group by field in its own column left of the detail columns.

OK

Applies the setting to convert the column and closes the dialog.

Cancel

Cancels the conversion and closes the dialog.

Help

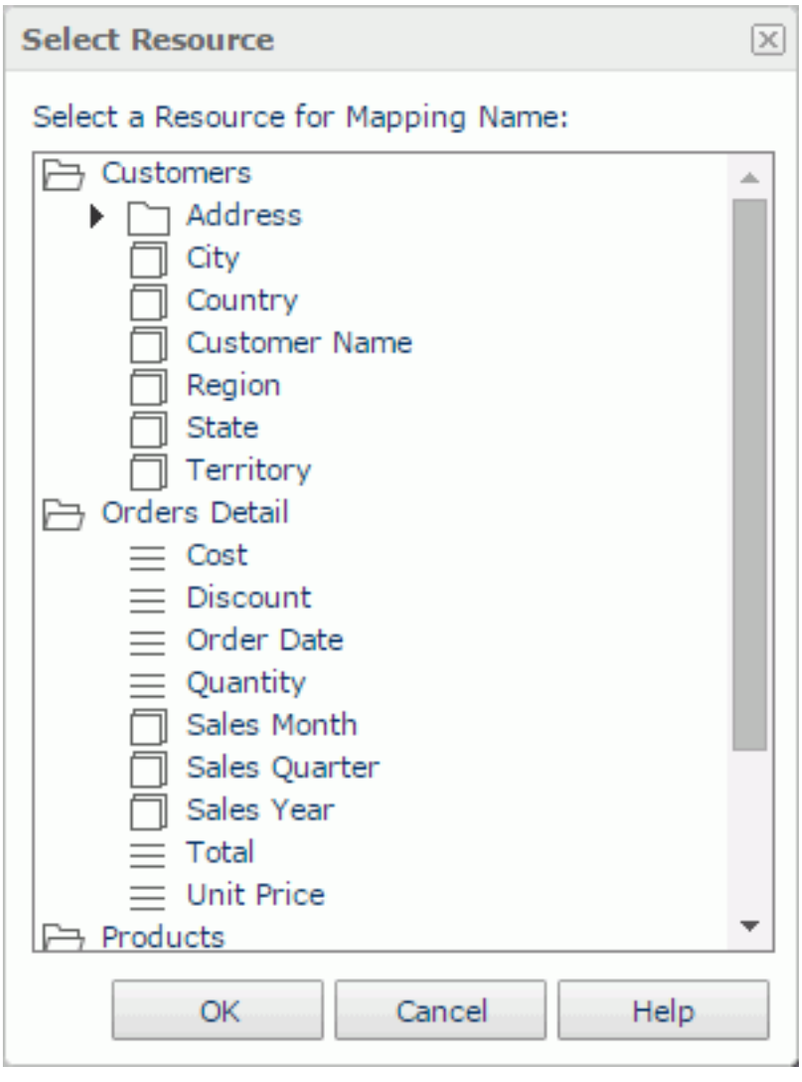
Displays the help document about this feature.

Related topics:

- Converting table columns

Select Resource dialog

This dialog helps you to specify the field to which the dynamic aggregation object will be mapped.



In this dialog, all the group objects and detail objects in the current business view and the dynamic formulas that have been created in the report will be listed. Select the required field and then click OK to make it as the mapping field of the aggregation object.

OK

Applies the settings and closes this dialog.

Cancel

Closes the dialog.

Help

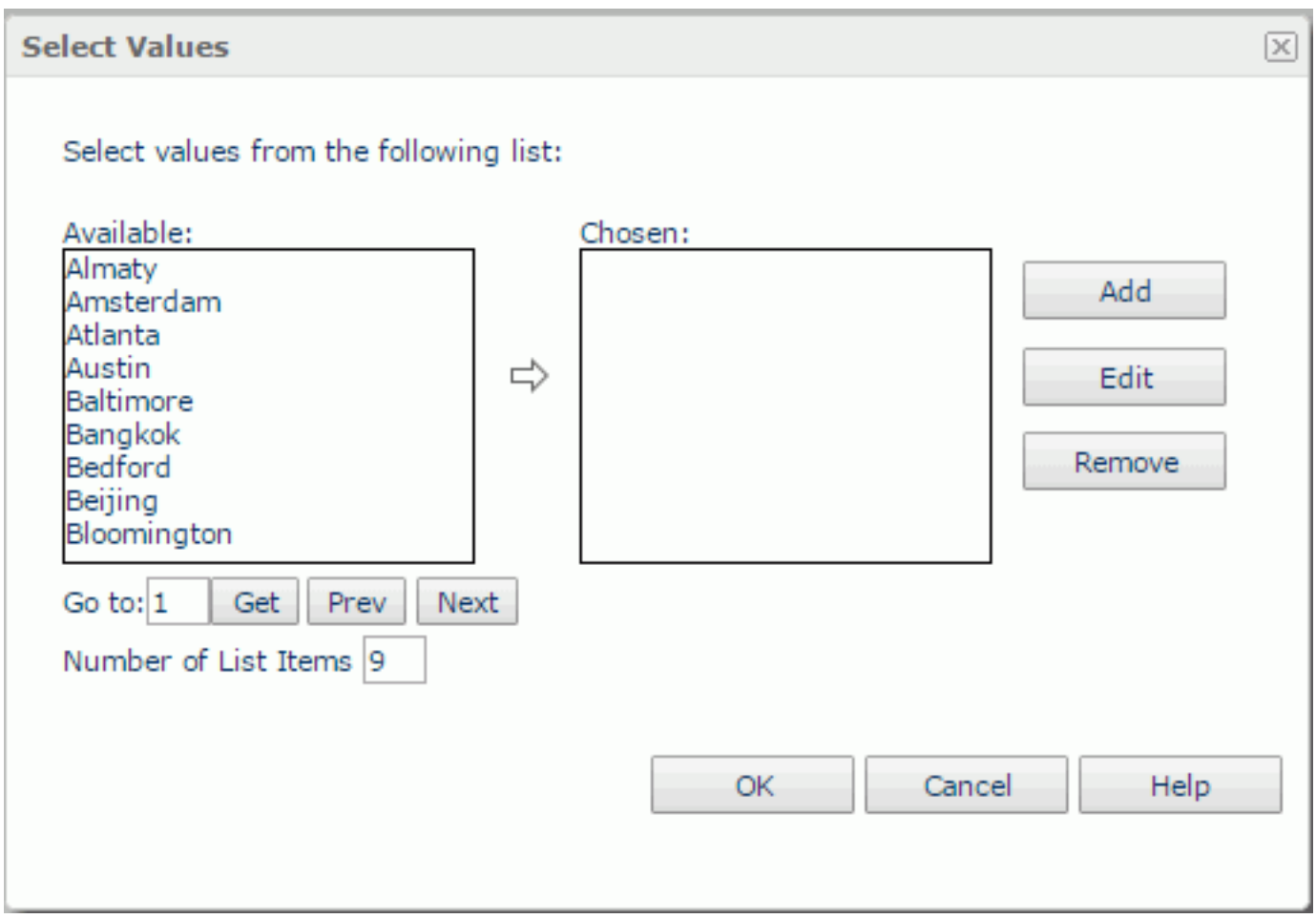
Displays the help document about this feature.

Related topics:

- Creating and using dynamic aggregation objects

Select Values dialog

This dialog helps you to specify one or more values to apply a filter criterion.



Available

Specifies the values from this list to filter report data with.

Adds the selected value to the Chosen list.

Chosen

Lists all the values that have been chosen.

Add

Adds a new line to the Chosen list for you to input a value.

Edit

Edits the selected item in the Chosen list. You can also double-click the item to edit it.

Remove

Removes the selected item from the Chosen list.

Go to

Goes to the specified value list. Input a number N here and click the Get button to fetch a value list starting from the Nth value.

Prev

Fetches the previous value list.

Next

Fetches the next value list.

Number of List Items

Specifies the number of list items here and click the Get button to fetch a value list.

OK

Applies all changes and closes this dialog.

Cancel

Closes the dialog and deletes any unsaved changes.

Help

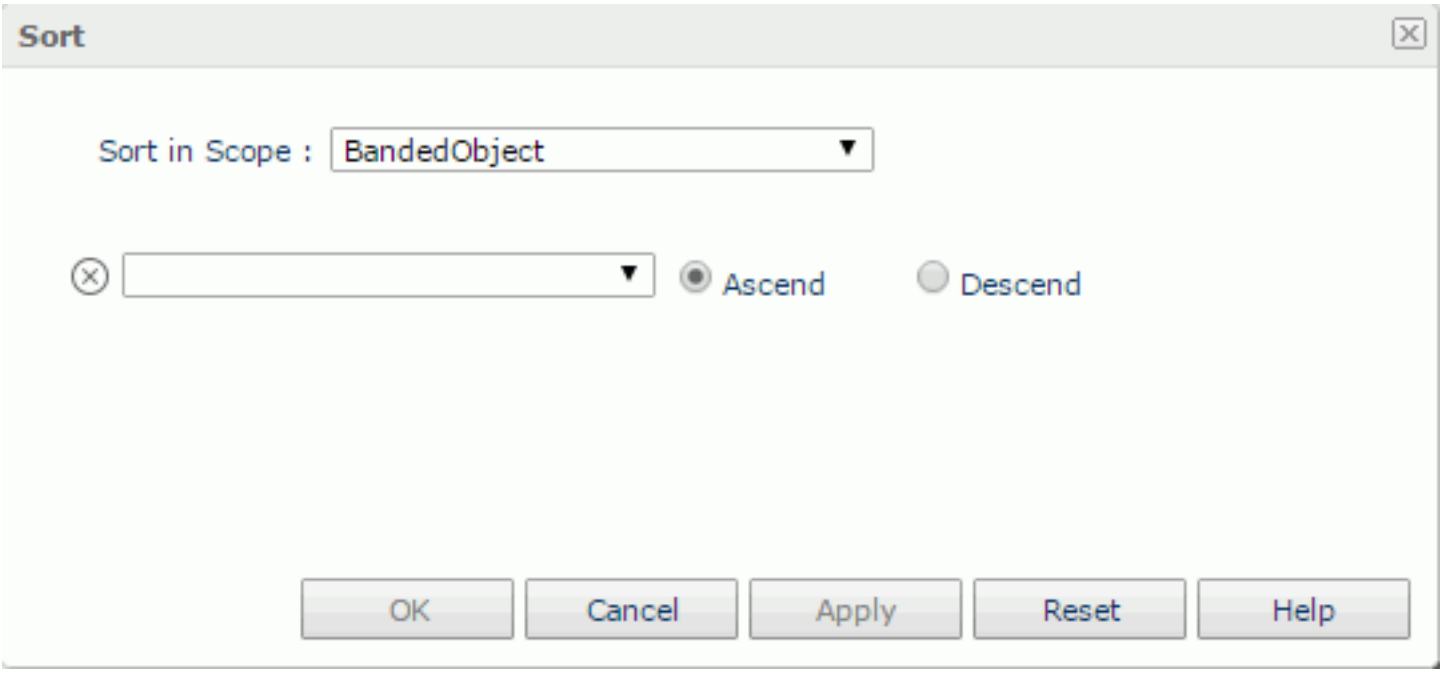
Displays the help document about this feature.

Related topics:

- Filtering the data components in a report

Sort dialog

This dialog helps you to specify the settings in order to sort the records or groups in a banded object.



Sort in Scope

Specifies on which level the sorting is based.

Deletes the specified sort expression line.

Field

Specifies the field by which to sort the records or groups.

Ascend

Specifies to sort the data ascendingly.

Descend

Specifies to sort the data descendingly.

Moves up a sort expression line.

Moves down a sort expression line.

Adds a new sort expression line after the current one. This button appears after the first sort condition has been composed.

OK

Accepts the changes and re-loads the report result using the sort settings with this dialog closed.

Cancel

Cancels the operation and closes this dialog.

Apply

Accepts the changes and re-loads the report result using the sort settings with this dialog displayed.

Reset

Restores this dialog to its opening status.

Help

Displays the help document about this feature.

Related topics:

- Sorting report data

Special Field Properties dialog

This dialog helps you to edit the properties of a special field. It contains the following tabs:

- General tab
- Font tab
- Border tab
- Others tab
- Display tab

OK

Applies the settings and closes this dialog.

Cancel

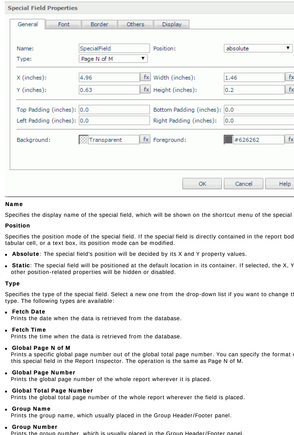
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the special field.



Name

Specifies the display name of the special field, which will be shown on the shortcut menu of the special field.

Position

Specifies the position mode of the special field. If the special field is directly contained in the report body, a tabular cell, or a text box, its position mode can be modified.

- **Absolute** The special field's position will be decided by its X and Y property values.
- **Static**. The special field will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled.

Type

Specifies the type of the special field. Select a new one from the drop-down list if you want to change the type. The following types are available:

- **Fetch Date**
  Prints the date when the data is retrieved from the database.
- **Fetch Time**
  Prints the time when the data is retrieved from the database.
- **Global Page N of M**
  Prints a specific global page number out of the global total page number. You can specify the format of this special field in the Report Inspector. The operation is the same as Page N of M.
- **Global Page Number**
  Prints the global page number of the whole report wherever it is placed.
- **Global Total Page Number**
  Prints the global total page number of the whole report wherever the field is placed.
- **Group Name**
  Prints the group name, which usually placed in the Group Header/Footer panel.
- **Group Number**
  Prints the group number, which is usually placed in the Group Header/Footer panel.
- **Modified Date**
  Prints the last modified date for the catalog.
- **Modified Time**
  Prints the last modified time for the catalog.
- **Page N of M**
  Prints the page number of the total page number.
- **Page Number**
  Prints the page number for each page.
- **Print Date**
  Prints today's date (or the date designated on your computer).
- **Print Time**
  Prints the current time on your computer.
- **Record Number**
  Prints the record number, which is usually placed in the Details panel.
- **SQL Statement**
  Prints the SQL statements used to execute the query.
- **Task ID**
  Prints the internal task ID, which is a unique time stamp.
- **Total Fetched Records**
  Prints the total number of records which take part in grouping calculation. The possible result of the special field is as follows:
  - If you don't set any filter condition in the Filter dialog, print the very number of the record obtained after setting the property Maximum Records.
  - If you set filter conditions in the Filter dialog, print the number of records obtained after performing the filters, even though you have set the property Maximum Records before setting the filters.
- **Total Group Number**
  Prints the total group number, which is usually placed in the group Header/Footer panel.
- **Total Page Number**
  Prints the total number of pages in the report.
- **Total Records**
  Prints the total number of records after all the filter conditions are performed, except the ones created in the Filter dialog of Page Report Studio, and the Group Filter dialog and top N or bottom N condition in JReport Designer.
- **User Name**
  Prints the User name with which you log onto JReport Server.

X

Specifies the X coordinate of the special field.

Y

Specifies the Y coordinate of the special field.

Width

Specifies the width of the special field.

Height

Specifies the height of the special field.

Background

Specifies the background color of the special field.

To change the color, click the color indicator to bring out the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

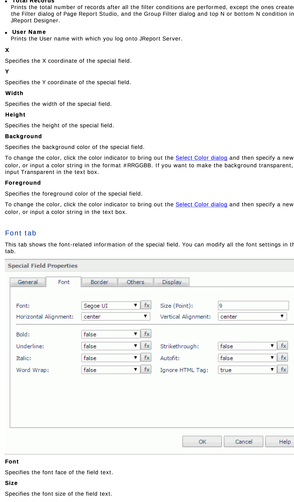
Foreground

Specifies the foreground color of the special field.

To change the color, click the color indicator to bring out the Select Color dialog and then specify a new color, or input a color string in the text box.

Font tab

This tab shows the font-related information of the special field. You can modify all the font settings in this tab.



Font

Specifies the font face of the field text.

Size

Specifies the font size of the field text.

Horizontal Alignment

Specifies the horizontal alignment mode of the text in the special field.

Vertical Alignment

Specifies the vertical alignment mode of the text in the special field.

Bold

Specifies whether to make the object text bold or not.

Underline

Specifies whether the field text will be underlined or not.

Strikethrough

Specifies whether or not to attach a strikeout line to the field text.

Italic

Specifies whether to make the field text italic or not.

Autofit

Specifies whether or not to automatically adjust the width of the special field.

Word Wrap

Specifies whether or not to wrap the text to the special field width.

Ignore HTML Tag

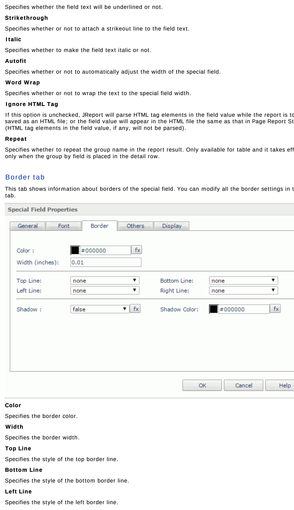
If this option is unchecked, JReport will parse HTML tag elements in the field value while the report is to be saved as an HTML file; or the field value will appear in the HTML file the same as that in Page Report Studio (HTML tag elements in the field value, if any, will not be parsed).

Repeat

Specifies whether to repeat the group name in the report result. Only available for table and it takes effect only when the group by field is placed in the detail row.

Border tab

This tab shows information about borders of the special field. You can modify all the border settings in this tab.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Shadow

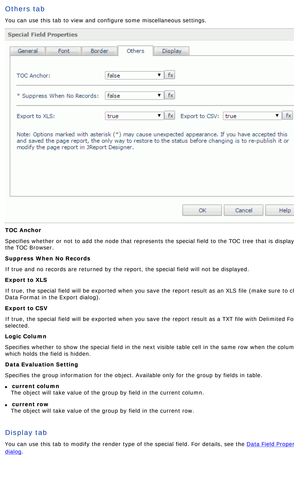
Specifies whether the borders will have a shadow effect or not.

Shadow Color

Specifies the color of the border shadow.

Others tab

You can use this tab to view and configure some miscellaneous settings.



TOC Anchor

Specifies whether or not to add the node that represents the special field to the TOC tree that is displayed in the TOC Browser.

Suppress When No Records

If true and no records are returned by the report, the special field will not be displayed.

Export to XLS

If true, the special field will be exported when you save the report result as an XLS file (make sure to check Data Format in the Export dialog).

Export to CSV

If true, the special field will be exported when you save the report result as a TXT file with Delimited Format selected.

Logic Column

Specifies whether to show the special field in the next visible table cell in the same row when the column which holds the field is hidden.

Data Evaluation Setting

Specifies the group information for the object. Available only for the group by fields in table.

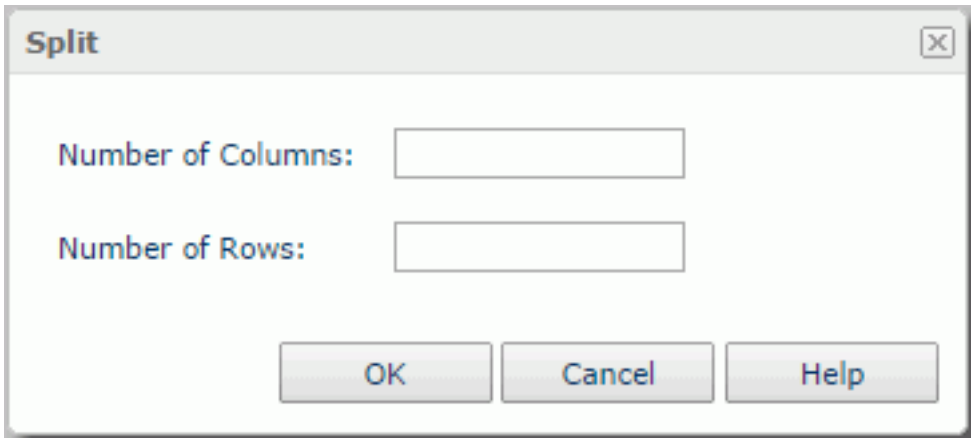
- **current column**
  The object will take value of the group by field in the current column.
- **current row**
  The object will take value of the group by field in the current row.

Display tab

You can use this tab to modify the render type of the special field. For details, see the Data Field Properties dialog.

Split dialog

This dialog helps you to specify the number of rows and columns that the specified tabular cell will be split into.



Number of Columns

Specifies the number of columns the tabular cell will be split into. Input a positive integer in the text box to specify the number.

Number of Rows

Specifies the number of rows the tabular cell will be split into. Input a positive integer in the text box to specify the number.

OK

Closes this dialog and applies the settings.

Cancel

Discards any modifications and closes this dialog.

Help

Displays the help document about this feature.

Related topics:

- Splitting and merging cells in a tabular

Table Cell Properties dialog

This dialog helps you to edit the properties of a table cell. It contains the following tabs:

- General tab
- Border tab
- Others tab

OK

Applies the settings and closes this dialog.

Cancel

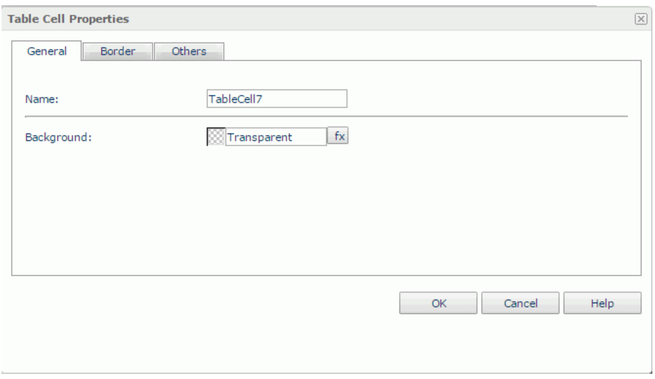
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the table cell.



Name

Specifies the display name of the table cell, which will be shown on the shortcut menu of the table cell.

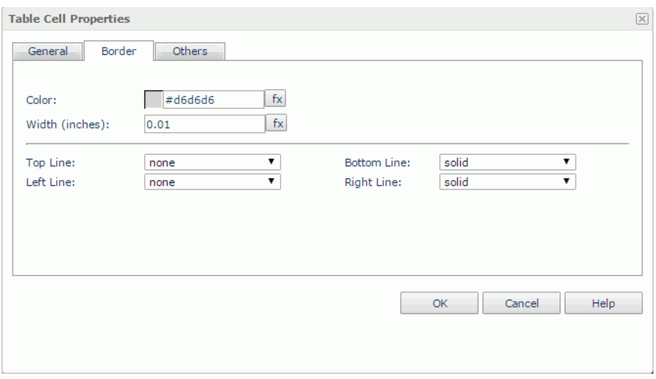
Background

Specifies the background color of the table cell.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the table cell. You can modify all the border settings in this tab.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

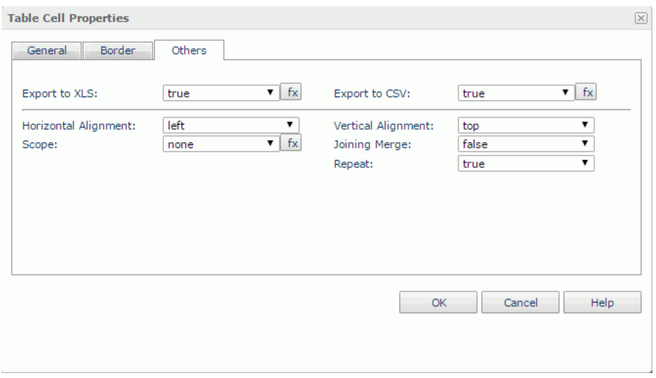
Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Others tab

You can use this tab to view and configure some miscellaneous settings.



Export to XLS

If true, the table cell will be exported when you save the report result as an XLS file (make sure to check Data Format in the Export dialog).

Export to CSV

If true, the table cell will be exported when you save the report result as a TXT file with Delimited Format selected.

Horizontal Alignment

Specifies the horizontal alignment mode of the content in the table cell. When the Position property for the object in the cell is set to absolute, this property does not take effect.

Vertical Alignment

Specifies the vertical alignment mode of the content in the table cell. When the Position property for the object in the cell is set to absolute, this property does note take effect.

Scope

A representation of the standard HTML attribute *scope*. This attribute specifies the set of data cells for which the current header cell provides header information.

- **Row** - The current cell provides header information for the rest of the row that contains it.
- **Column** - The current cell provides header information for the rest of the column that contains it.
- **None** - The scope attribute will not be generated when exporting to HTML.

Joining Merge

Specifies whether to make all the sequential rows included in the cell vertically merged in the report result. It is meaningful to set the property only when the cell is merged with other cells in the same column.

Repeat

Specifies whether to repeat the content of the cell in every page of the report result when the cell is split into pages. Available only for vertically merged cell.

Table Properties dialog

This dialog helps you to edit the properties of a table. It contains the following tabs:

- General tab
- Border tab
- Others tab

OK

Applies the settings and closes this dialog.

Cancel

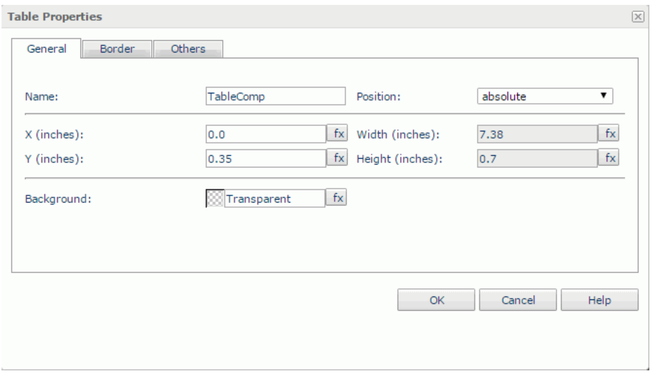
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the table.



Name

Specifies the display name of the table, which will be shown on the shortcut menu of the table.

Position

Specifies the position mode of the table. If the table is directly contained in the report body, a tabular cell, or a text box, its position mode can be modified.

- **Absolute**: The table's position will be decided by its X and Y property values.
- **Static**: The table will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled.

X

Specifies the X coordinate of the table.

Y

Specifies the Y coordinate of the table.

Width

Specifies the width of the table.

Height

Specifies the height of the table.

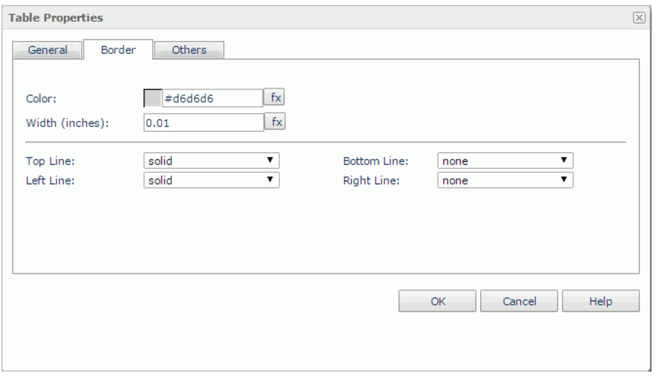
Background

Specifies the background color of the table.

To change the color, click the color indicator to bring out the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the table.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

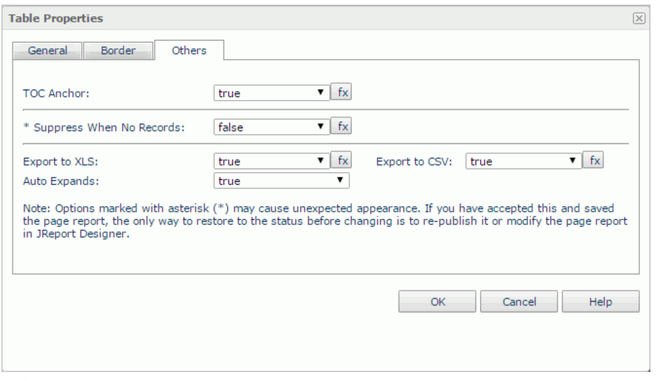
Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Others tab

You can use this tab to view and configure some miscellaneous settings.



TOC Anchor

Specifies whether or not to add the node that represents the table to the TOC tree that is displayed in the TOC Browser.

Suppress When No Records

If true and no records are returned by the report, the table will not be displayed.

Export to XLS

If true, the table will be exported when you save the report result as an XLS file (make sure to check Data Format in the Export dialog).

Export to CSV

If true, the table will be exported when you save the report result as a TXT file with Delimited Format selected.

Table Row Properties dialog

This dialog helps you to edit the properties of a table row. It contains the following tabs:

- General tab
- Others tab

OK

Applies the settings and closes this dialog.

Cancel

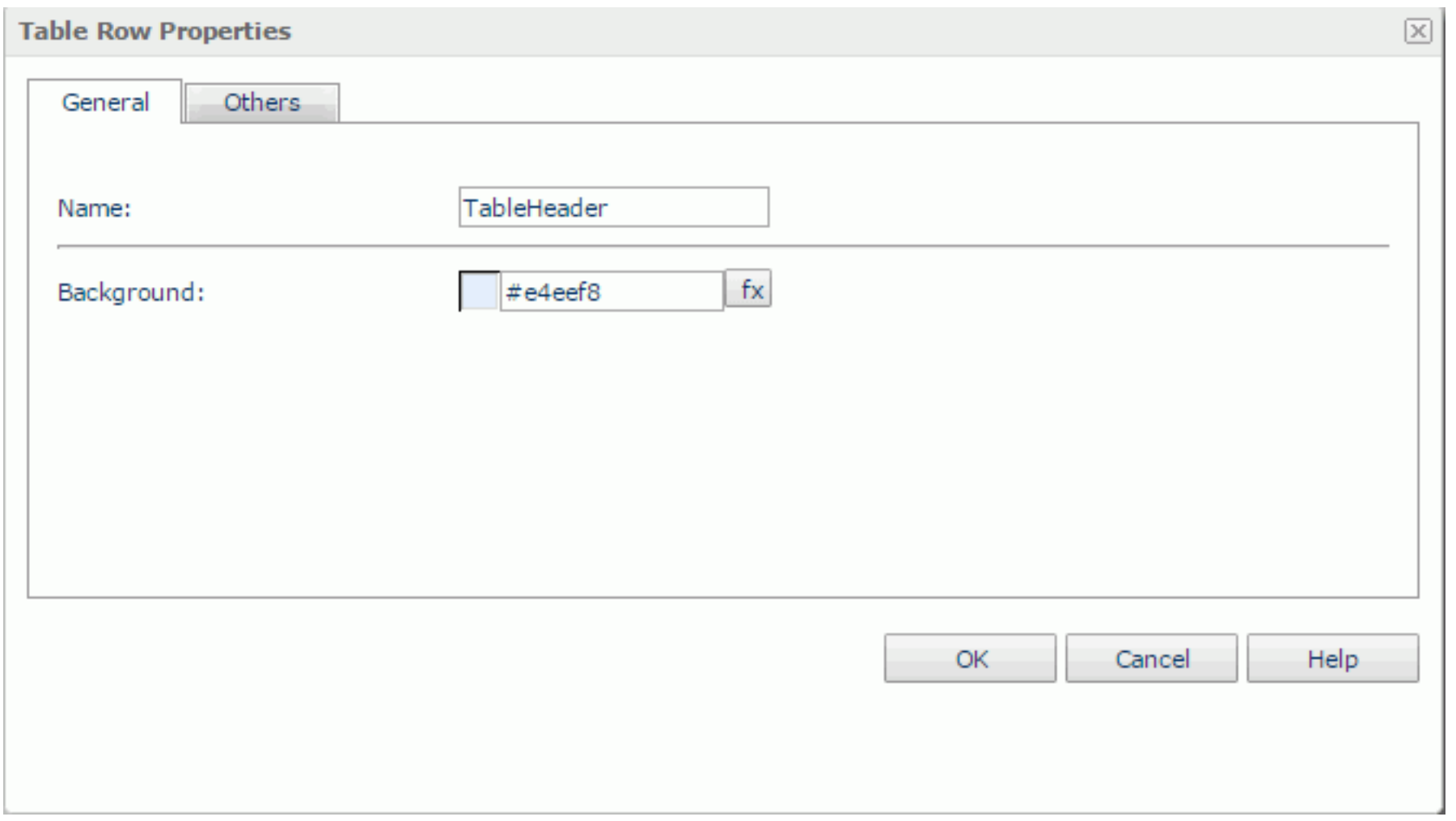
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the object.



Name

Specifies the display name of the table row, which will be shown on its shortcut menu.

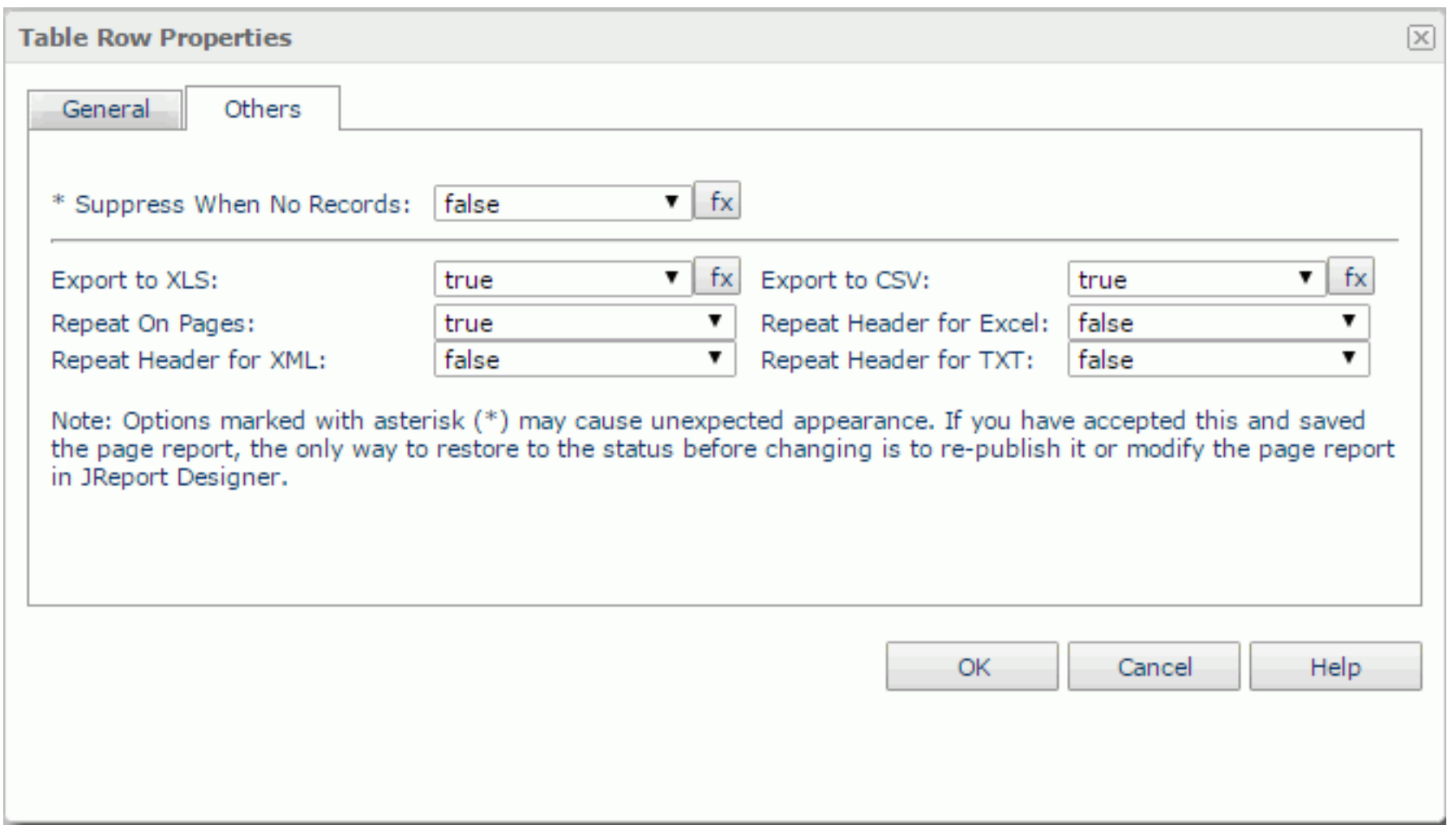
Background

Specifies the background color of the table row.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Others tab

You can use this tab to view and configure some miscellaneous settings.



Suppress When No Records

If true and no records are returned by the report, the table row will not be displayed.

Export to XLS

If true, the table row will be exported when you save the report result as an XLS file (make sure to check Data Format in the Export dialog).

Export to CSV

If true, the table row will be exported when you save the report result as a TXT file with Delimited Format selected.

Table Wizard

This wizard guides you through the process of creating a table report. It contains the following screens:

- Data screen
- Display screen
- Group screen
- Summary screen
- Query Filter screen
- Style screen

Back

Returns to the previous screen.

Next

Goes to the next screen.

Finish

Creates a report containing the table and closes the wizard.

Cancel

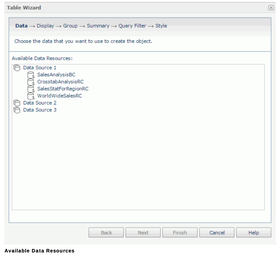
Closes the wizard without creating a report.

Help

Displays the help document about this feature.

Data screen

Specifies the business view to use to create the table. This screen is hidden when there is only one business view in the current catalog.

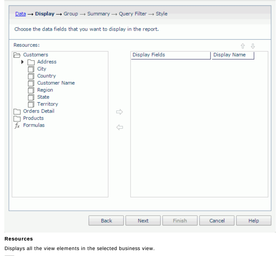


Available Data Resources

Lists all the available business views in the current catalog, with which you can create the table.

Display screen

Specifies the fields to be displayed in the table.



Resources

Displays all the view elements in the selected business view.

Adds the selected view element to be displayed in the table.

Removes the selected view element.

Display Fields

Lists the view elements that have been added to the table.

Display Name

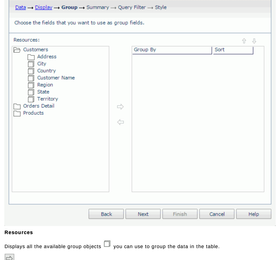
Specifies the display names of the added view elements.

Moves the selected view element one step up.

Moves the selected view element one step down.

Group screen

Specifies the fields to group the data.



Resources

Displays all the available group objects you can use to group the data in the table.

Adds the selected group object as a group field.

Removes the selected group object.

Group By

Lists all the group objects that have been added as group fields.

Sort

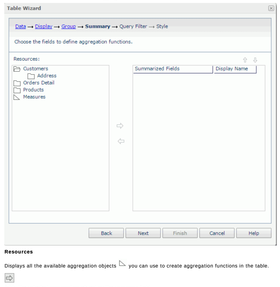
Specifies the sort order for each group: Ascend, Descend, or No Sort.

Moves the selected group one step up.

Moves the selected group one step down.

Summary screen

Specifies the fields on which to create aggregation functions.



Resources

Displays all the available aggregation objects you can use to create aggregation functions in the table.

Adds the selected aggregation object as the summary field.

Removes the selected aggregation object.

Summarized Fields

Lists the groups that have been created in the table and the aggregation objects you have added to summarize data in each group.

Display Name

Specifies the display names of the aggregation objects.

Row

Specifies to put the summary field in the header or footer row. If the summary is calculated on a group by field, it will be put in the group header or footer row of the corresponding group; if the summary is calculated on the whole dataset, it will be put in the table header or footer row. Available only when the table is Group Left type.

Column

Specifies to put the summary field in the specified detail column. Available only when the table is Group Left type.

Moves the selected aggregation object one step up.

Moves the selected aggregation object one step down.

Query Filter screen

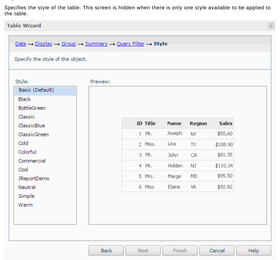
Specifies the filter which you want to apply to the selected business view.

In this screen, all the predefined filters of the business view are listed in the Query Filter drop-down list. You can choose one of them to apply. If you prefer to define a filter on your own, select **User Defined** from the drop-down list, and then define it according to your requirements.

For details about options in the screen, refer to Query Filter dialog.

Style screen

Specifies the style of the table. This screen is hidden when there is only one style available to be applied to the table.



Style

Lists all table styles for you to select one from.

Inherit Style

Specifies to take the style of the parent component. The option is available only when you specify to insert the table into a banded object.

Preview

Shows a preview of the selected style.

Related topics:

- Creating a table report

Tabular Cell Properties dialog

This dialog helps you to edit the properties of a tabular cell. It contains the following tabs:

- General tab
- Border tab
- Others tab

OK

Applies the settings and closes this dialog.

Cancel

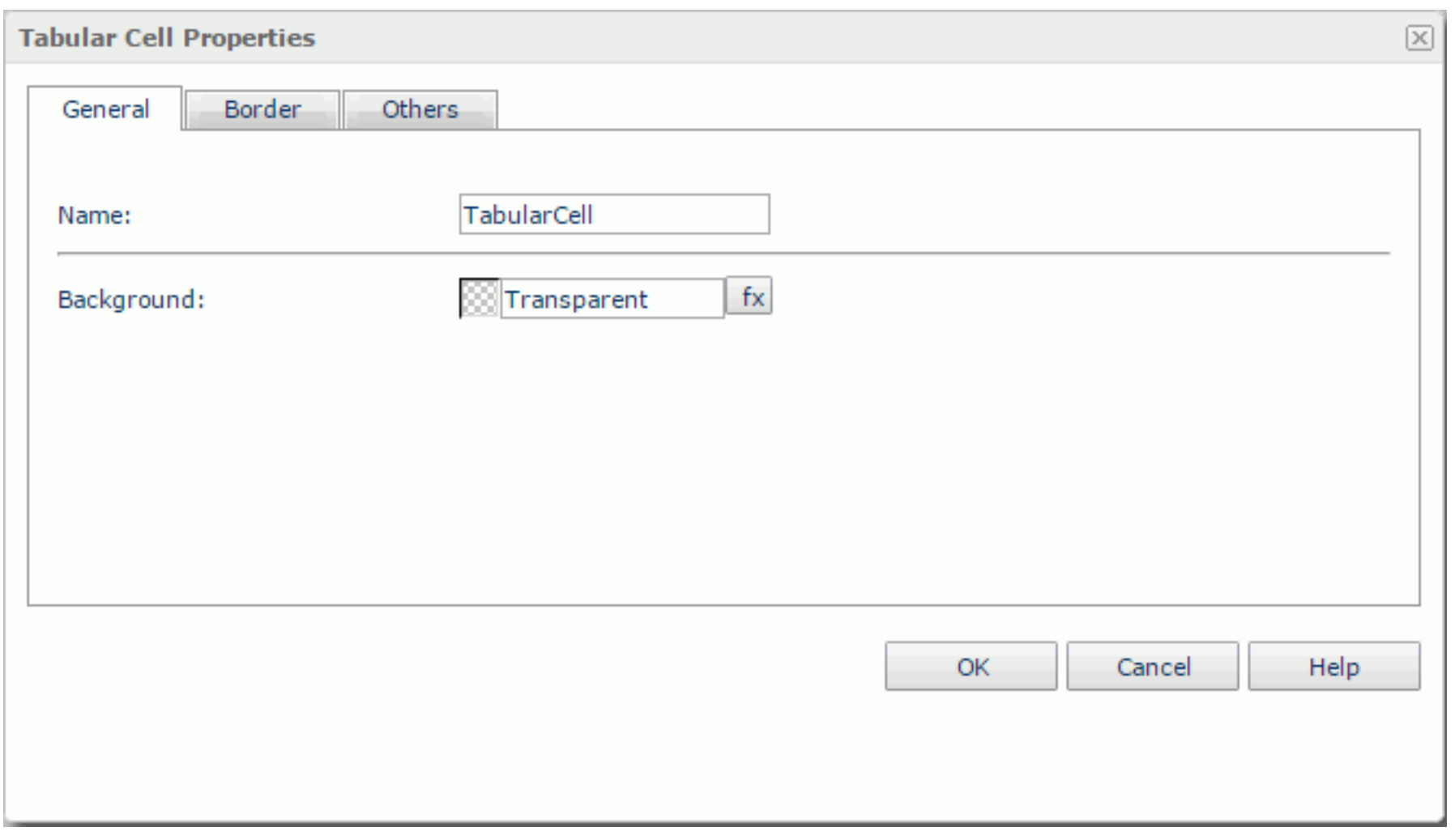
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the tabular cell.



Name

Specifies the display name of the tabular cell, which will be shown on its shortcut menu.

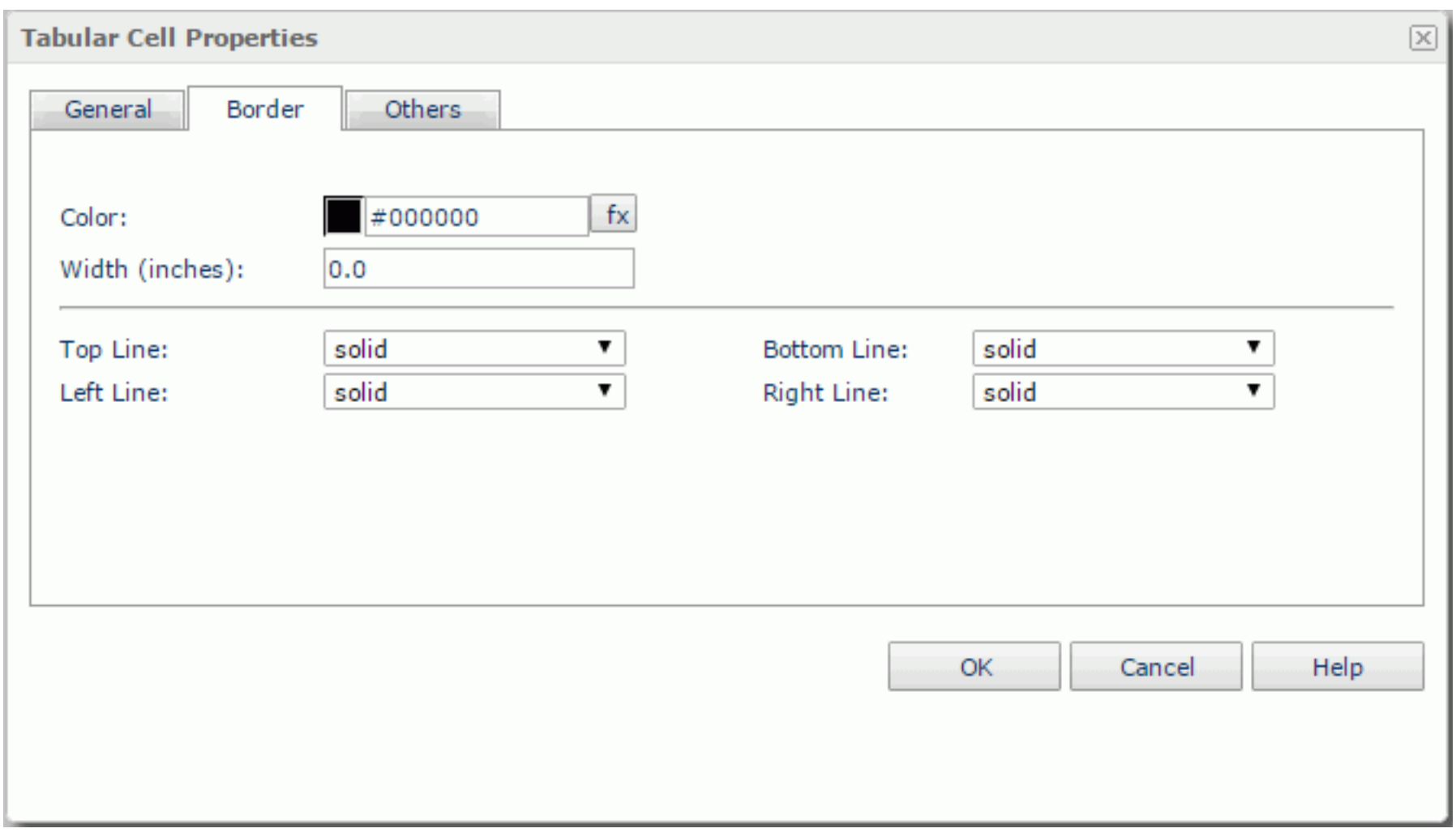
Background

Specifies the background color of the tabular cell.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the tabular cell.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

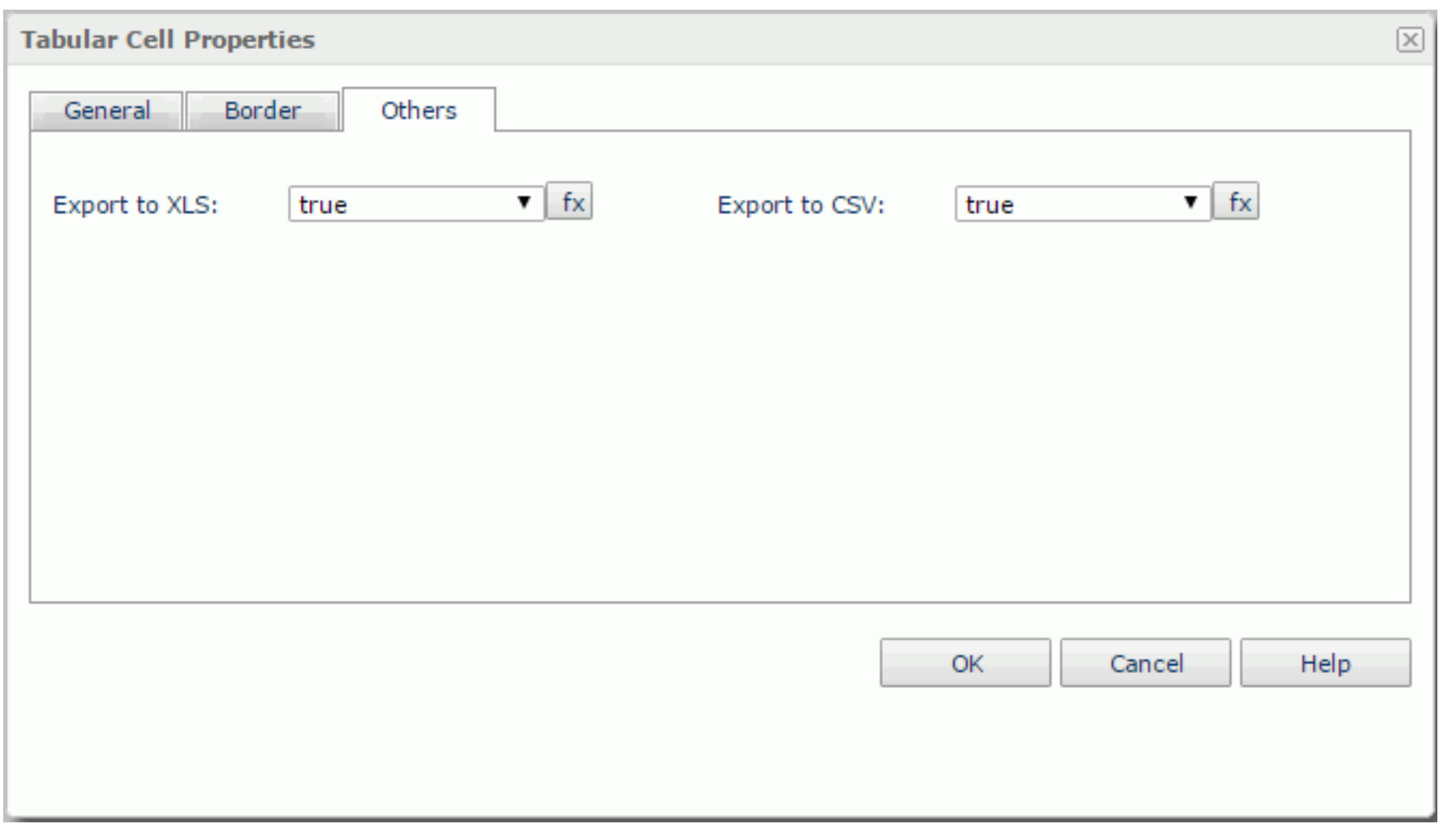
Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Others tab

You can use this tab to view and configure some miscellaneous settings.



Export to XLS

If true, the tabular cell will be exported when you save the report result as an XLS file (make sure to check Data Format in the Export dialog).

Export to CSV

If true, the tabular cell will be exported when you save the report result as a TXT file with Delimited Format selected.

Tabular Properties dialog

This dialog helps you to edit the properties of a tabular. It contains the following tabs:

- General tab
- Others tab

OK

Applies the settings and closes this dialog.

Cancel

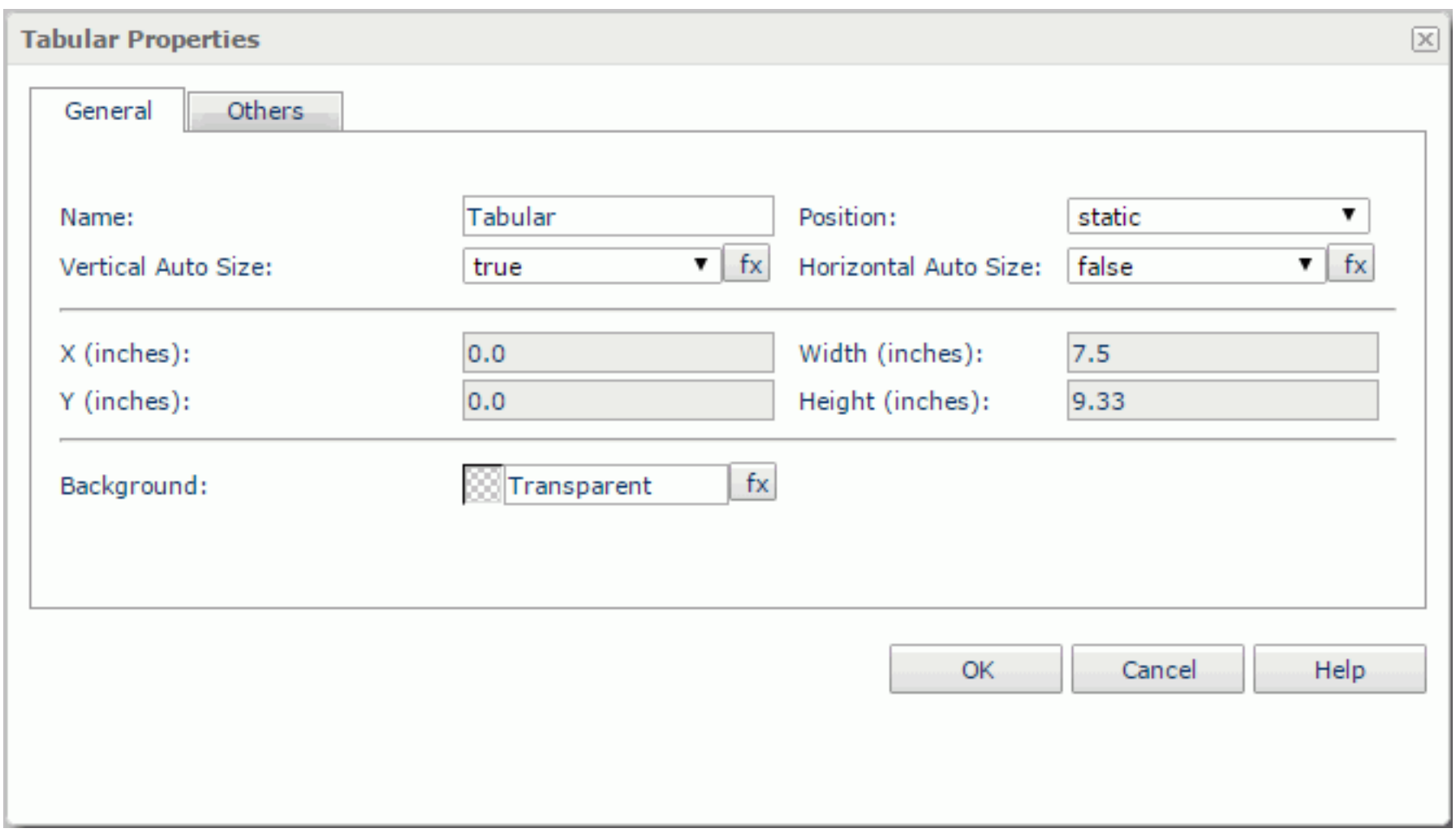
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the tabular.



Name

Specifies the display name of the tabular, which will be shown on the shortcut menu of the tabular.

Position

Specifies the position mode of the tabular. If the tabular is directly contained in the report body, a tabular cell, or a text box, its position mode can be modified.

- **Absolute**: The tabular's position will be decided by its X and Y property values.
- **Static**: The tabular will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled.

Vertical Auto Size

Specifies whether or not to automatically adjust the height of the tabular according to the size of the components inserted.

Horizontal Auto Size

Specifies whether or not to automatically adjust the width of the tabular according to the size of the components inserted.

X

Specifies the X coordinate of the tabular.

Y

Specifies the Y coordinate of the tabular.

Width

Specifies the width of the tabular.

Height

Specifies the height of the tabular.

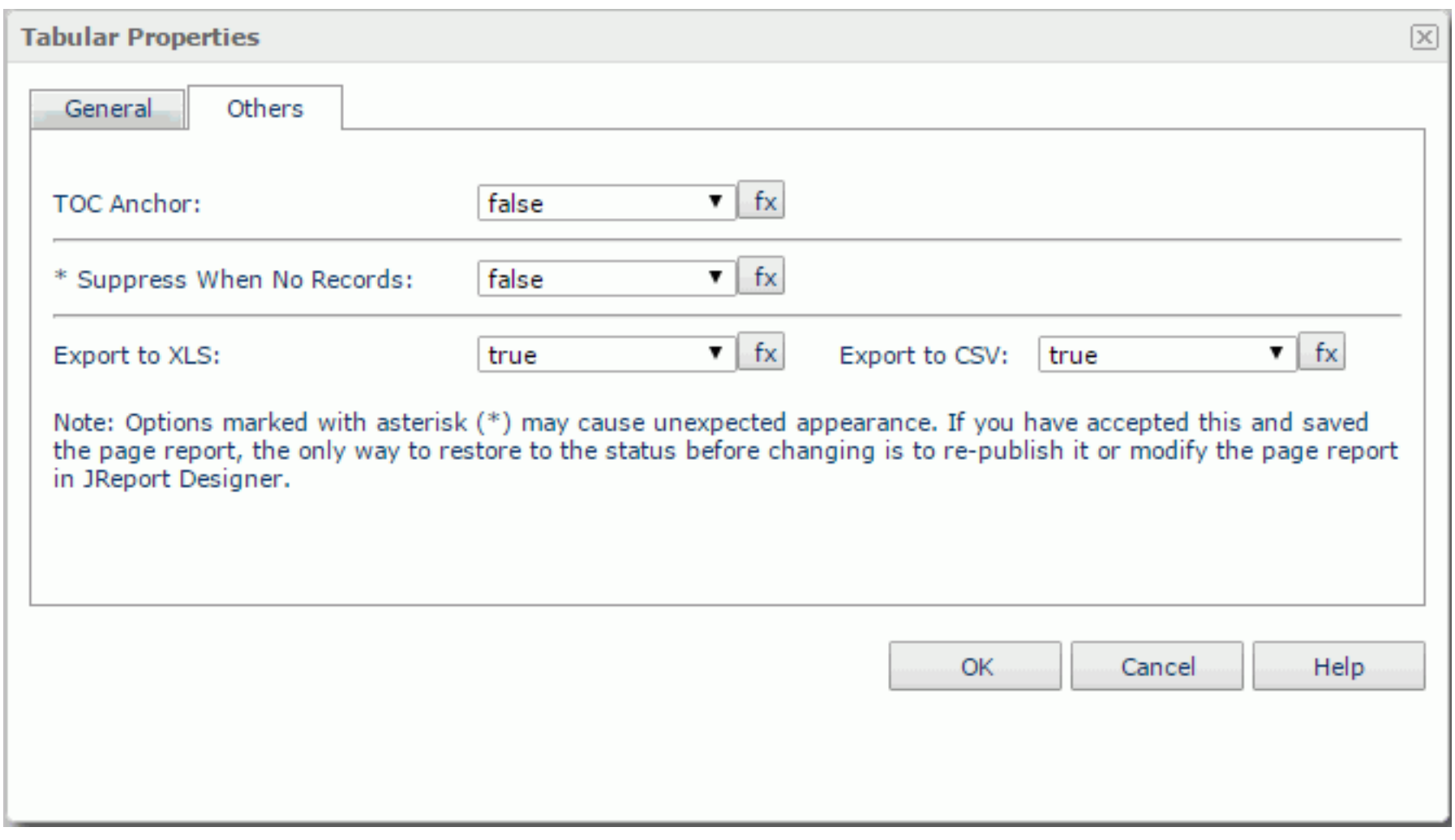
Background

Specifies the background color of the tabular.

To change the color, click the color indicator to access the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Others tab

You can use this tab to view and configure some miscellaneous settings.



TOC Anchor

Specifies whether or not to add the node that represents the tabular to the TOC tree that is displayed in the TOC Browser.

Suppress When No Records

If true and no records are returned by the report, the tabular will not be displayed.

Export to XLS

If true, the tabular will be exported when you save the report result as an XLS file (make sure to check Data Format in the Export dialog).

Export to CSV

If true, the tabular will be exported when you save the report result as a TXT file with Delimited Format selected.

Text Box Properties dialog

This dialog helps you to edit the properties of a text box. It contains the following tabs:

- General tab
- Border tab
- Others tab

OK

Applies the settings and closes this dialog.

Cancel

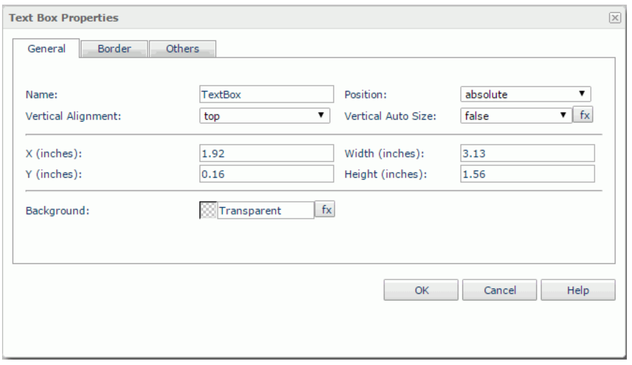
Cancels the settings and closes this dialog.

Help

Displays the help document about this feature.

General tab

This tab shows some general information of the text box.



Name

Specifies the display name of the text box, which will be shown on its shortcut menu.

Position

Specifies the position mode of the text box. If the text box is directly contained in the report body, a tabular cell, or a text box, its position mode can be modified.

- **Absolute**: The text box's position will be decided by its X and Y property values.
- **Static**: The text box will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled.

Vertical Alignment

Specifies vertical justification of the text in the text box.

Vertical Auto Size

Specifies whether or not to automatically adjust the height of the text box according to the size of the components inserted.

X

Specifies the X coordinate of the text box.

Y

Specifies the Y coordinate of the text box.

Width

Specifies the width of the text box.

Height

Shows the height of the text box.

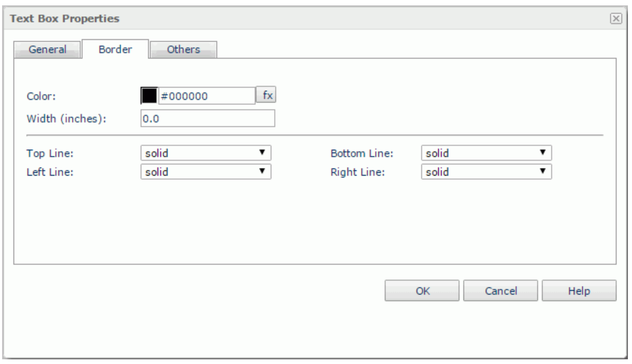
Background

Specifies the background color of the text box.

To change the color, click the color indicator to bring out the Select Color dialog and then specify a new color, or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the text box.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

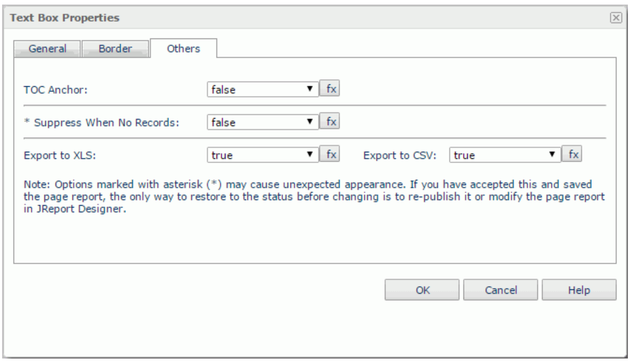
Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Others tab

You can use this tab to view and configure some miscellaneous settings.



TOC Anchor

Specifies whether or not to add the node that represents the text box to the TOC tree that is displayed in the TOC Browser.

Suppress When No Records

If true and no records are returned by the report, the text box will not be displayed.

Export to XLS

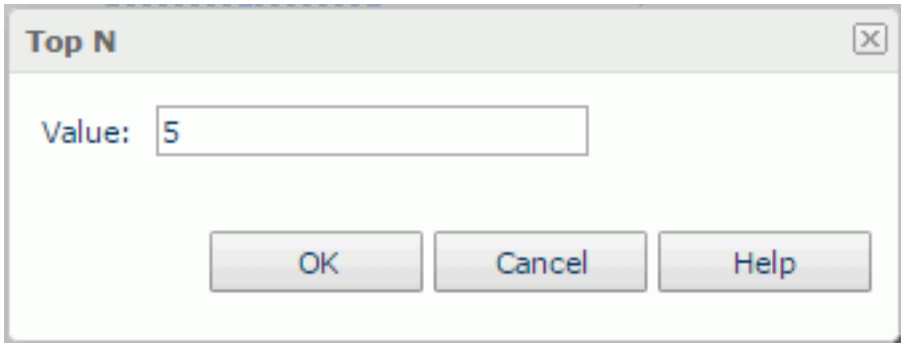
If true, the text box will be exported when you save the report result as an XLS file (make sure to check Data Format in the Export dialog).

Export to CSV

If true, the text box will be exported when you save the report result as a TXT file with Delimited Format selected.

Top N dialog

This dialog helps you to filter data to display records that meet the Top N condition.



Value

Specifies a positive integer N here so that records with the field value equal to one of the top N field values will be displayed.

OK

Applies the settings and closes the dialog.

Cancel

Closes the dialog and discards any changes.

Help

Displays the help document about this feature.

Related topics:

- Filtering the data components in a report

To Chart dialog

This dialog helps you to specify settings for converting a crosstab into a chart. It contains the following tabs:

- Chart Type tab
- Display tab
- Style tab

Back

Returns to the previous tab.

Next

Goes to the next tab.

OK

Applies the settings and closes the dialog.

Cancel

Cancels the settings and closes the dialog.

Help

Displays the help document about this feature.

Chart Type tab

This tab allows you to set the type of the chart.

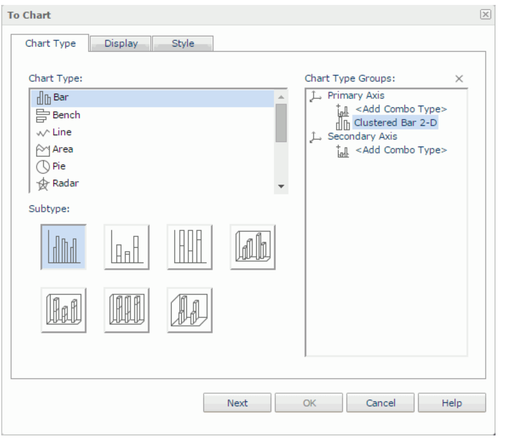


Chart Type

Lists all chart types for you to select one.

Subtype

Shows subtypes of the selected chart type in thumbnail form.

Chart Type Groups

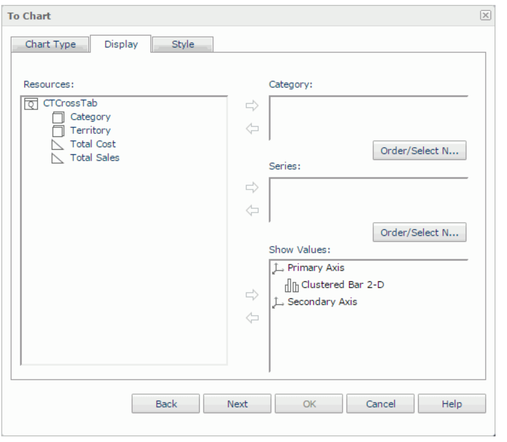
Lists the subtypes defined for the chart.

×

Removes the selected subtype.

Display tab

This tab allows you to set the fields that will be displayed in the chart.



Resources

Displays all the view elements used in the crosstab.

Adds the selected group or aggregation object to be displayed in the chart.

Removes the selected view element.

Category

Lists the group object that will be displayed on the category axis of the chart.

Series

Lists the group object that will be displayed on the series axis of the chart.

Show Values

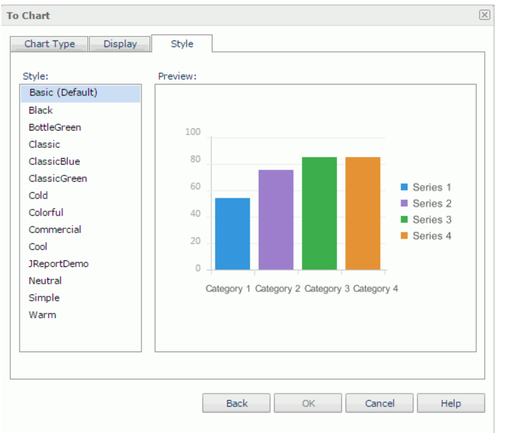
Lists the aggregation objects that will be displayed on the value axis of the chart.

Order/Select N

Opens the Order/Select N dialog to define the sort order and Select N condition in the chart.

Style tab

This tab allows you to select a style for the chart. It is hidden when there is only one style available.



Style

Lists all the styles for you to select from.

- **Custom**
  There is no style information in this style and it is only used to support reports built with previous versions which did not bind any style or the bound style cannot be found in the style list.

Preview

Shows a preview of the selected style.

Inherit Style

Specifies to take the style of the parent component. The option is available when the crosstab is in a banded object.

Related topics:

- Converting between components

To Crosstab dialog

This dialog helps you to specify settings for converting a chart into a crosstab. It contains the following tabs:

- Display tab
- Style tab

Back

Returns to the previous tab.

Next

Goes to the next tab.

OK

Applies the settings and closes the dialog.

Cancel

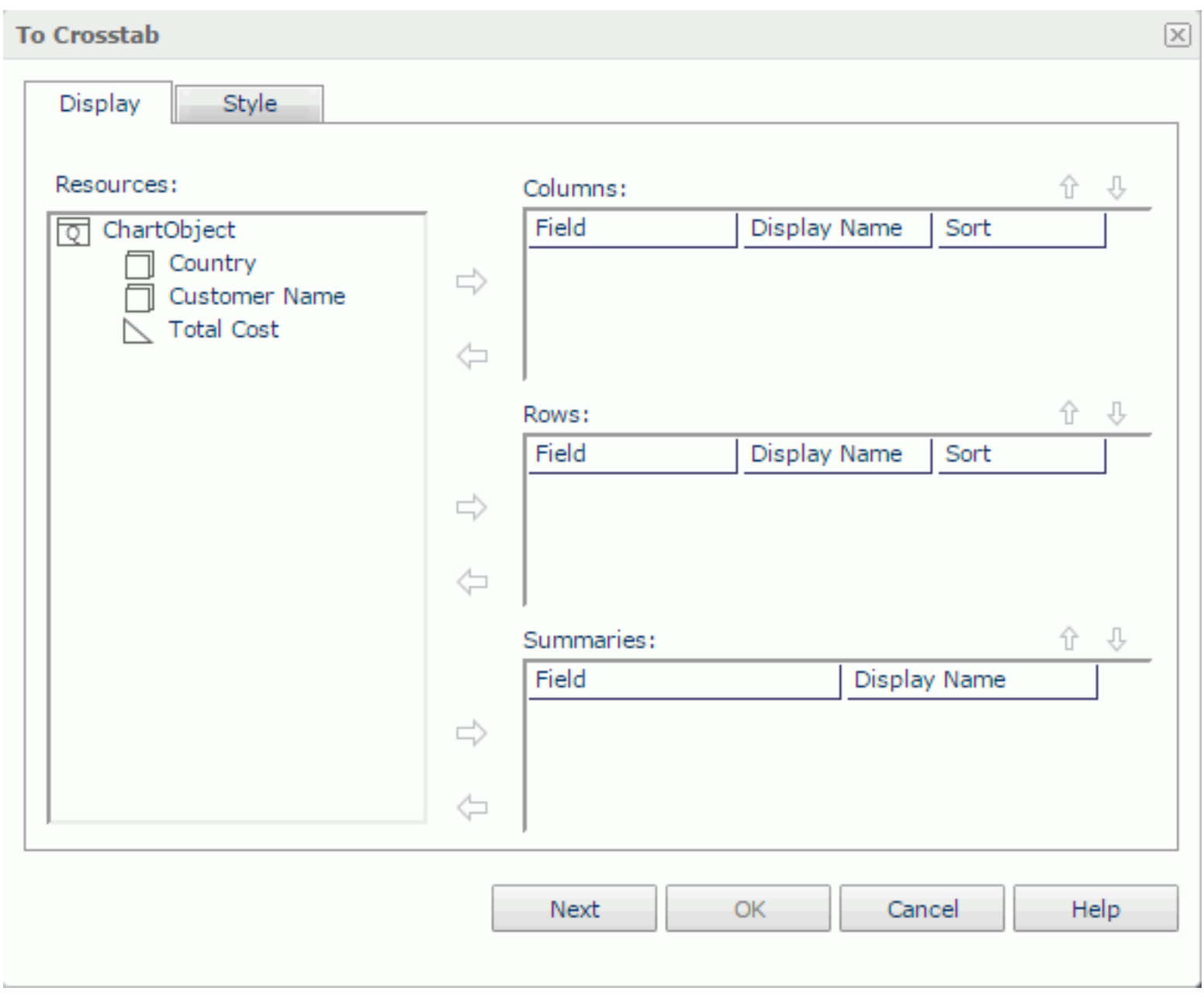
Cancels the settings and closes the dialog.

Help

Displays the help document about this feature.

Display tab

This tab allows you to set the data fields that you want to display in the crosstab.



Resources

Displays the view elements used in the chart.

Adds the selected view element to the crosstab.

Removes the selected view element.

Columns/Rows

Lists the group objects that will be displayed on the columns/rows of the crosstab.

- **Field**
  Lists the group objects that you selected to display in the crosstab.
- **Display Name**
  Lists the display name for the selected group objects. You can edit the names if required.
- **Sort**
  Specifies how the selected group objects will be sorted.

Summaries

Lists the aggregation objects that will be the aggregate fields of the crosstab.

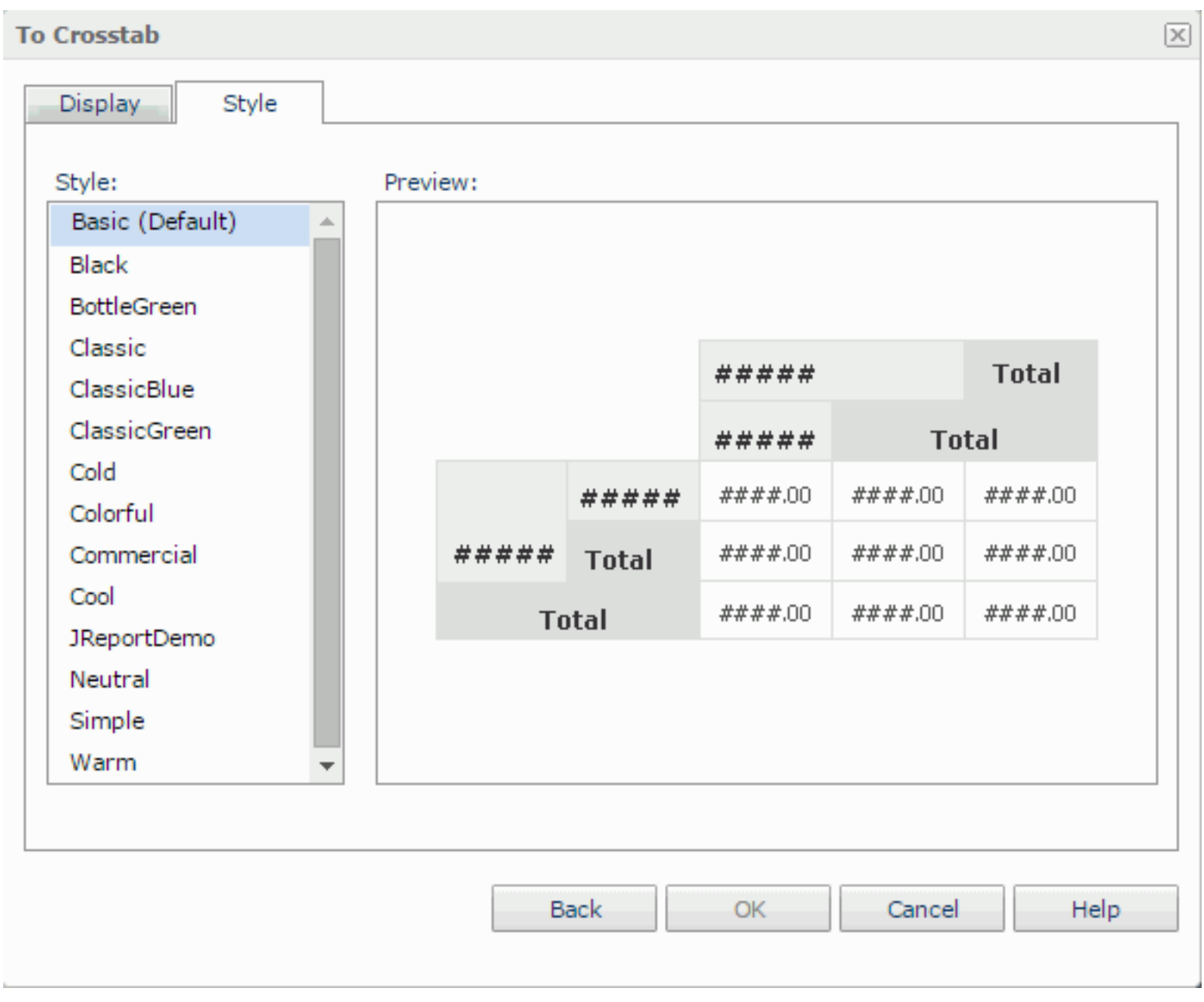
- **Field**
  Lists the aggregation objects that you selected to display in the crosstab.
- **Display Name**
  Lists the display name for the selected aggregation objects. You can edit the names if required.

Moves the selected view element one step up.

Moves the selected view element one step down.

Style tab

This tab allows you to select a style for the crosstab. It is hidden when there is only one style available.



Style

Lists all the styles for you to select one.

- **Custom**
  There is no style information in this style and it is only used to support reports built with previous versions which did not bind any style or the bound style cannot be found in the style list.

Preview

Shows a preview of the selected style.

Inherit Style

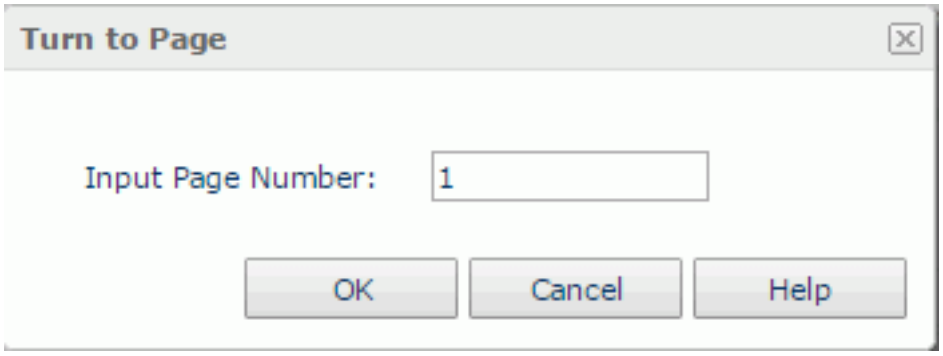
Specifies to take the style of the parent component. The option is available when the chart is in a table or banded object.

Related topics:

- Converting between components

Turn to Page dialog

This dialog helps you to input the page number so as to turn to the specified page.



Input Page Number

Specifies the page number you would like to turn to.

Go

Goes to the specified page.

Cancel

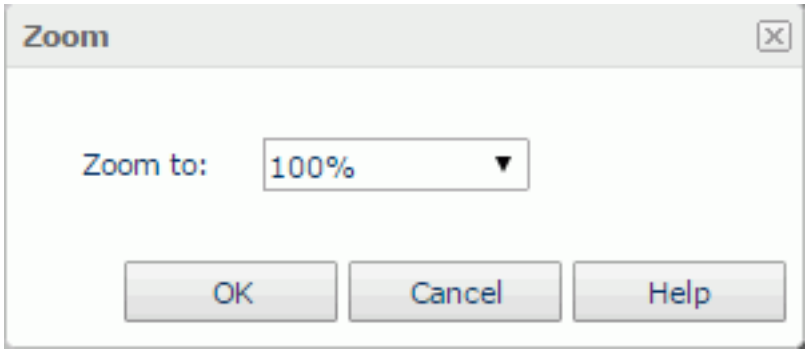
Closes this dialog without turning the report page.

Help

Displays the help document about this feature.

Zoom dialog

This dialog helps you to set a magnification of the report page.



Zoom to

Specifies the magnification in percentage.

OK

Closes this dialog and zooms the report page in or out to the magnification.

Cancel

Closes this dialog without changing the magnification of the report page.

Help

Displays the help document about this feature.

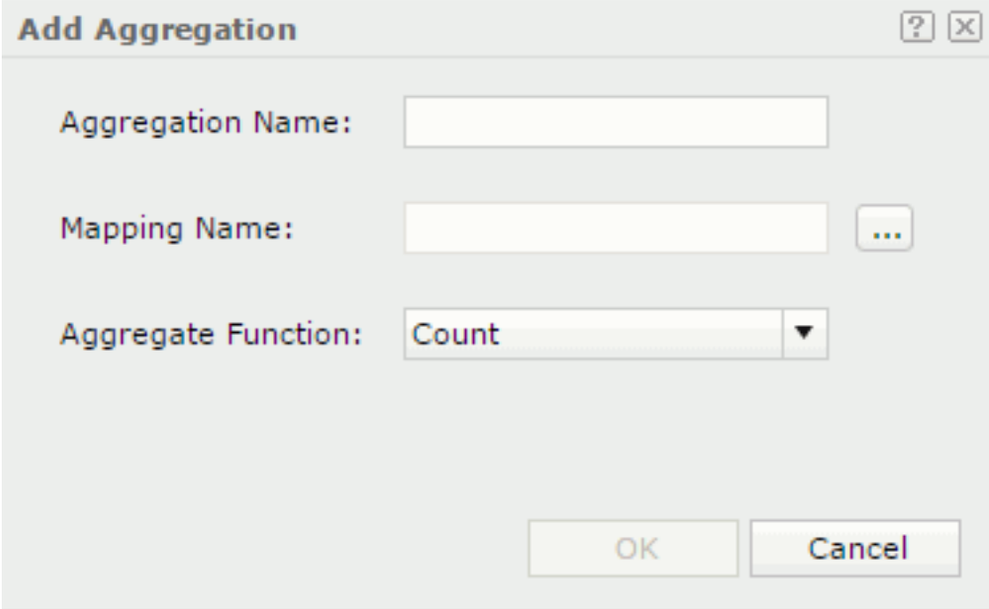
Web Report Studio dialogs

While using Web Report Studio, you will go through the following dialogs. This section introduces each of these dialogs in alphabetical order. You can go into the links for details.

- Add Aggregation dialog
- Add Parameter dialog
- Aggregate On dialog
- Bind Data dialog
- Button Properties dialog
- Category Options dialog
- Chart Properties dialog
- Chart Wizard
- Color Picker dialog
- Comparison Function dialog
- Conditional Formatting dialog
- Crosstab Properties dialog
- Crosstab Wizard
- Custom Sort dialog
- Data Format dialog
- Edit Additional Value dialog
- Edit Conditions dialog
- Edit Detail Table dialog
- Edit Filter Control dialog
- Edit Image dialog
- Edit Link dialog
- Edit Multimedia dialog
- Edit Parameter dialog
- Enter Values dialog
- Export dialog
- Field Properties dialog
- Fill Effects dialog
- Filter Control Properties dialog
- Filter dialog
- Filter Inspector dialog
- Flash Properties dialog
- Font dialog
- Format Category (X) Axis dialog
- Format Category (X) Gridline dialog
- Format Graph dialog
- Format Legend dialog
- Format Org Line dialog
- Format Org Node dialog
- Format Paper dialog
- Format Platform dialog
- Format Rectangle dialog
- Format Rectangle Title dialog
- Format Series (Z) Gridline dialog
- Format Value (Y) Axis dialog
- Format Value (Y) Gridline dialog
- Format Value (Y2) Axis dialog
- Format Value (Y2) Gridline dialog
- Format Wall dialog
- Formula Editor dialog
- Group Footer Properties dialog
- Group Header Properties dialog
- Group Options dialog
- Image Properties dialog
- Insert Aggregation dialog
- Insert Chart dialog
- Insert Crosstab dialog
- Insert Filter Control dialog
- Insert Image dialog
- Insert Link dialog
- Insert Multimedia dialog
- Insert Parameter Control dialog
- Insert Parameter Form Control dialog
- Insert Table dialog
- Label Properties dialog
- Navigation Control Properties dialog
- Page Setup dialog
- Parameter Control Properties dialog
- Parameter Field Properties dialog
- Parameter Form Control Properties dialog
- PDF Encrypt Option dialog
- PDF Sign Option dialog
- Print dialog
- Printer Properties dialog
- Query Filter dialog
- RealMedia Properties dialog
- Report Body Properties dialog
- Report Parameters dialog
- Save As dialog
- Select a Report dialog
- Select Field dialog
- Select Resource dialog
- Select Value dialog
- Select Values dialog
- Series Options dialog
- Show Objects dialog
- Special Field Properties dialog
- Split Cell dialog
- Summary Properties dialog
- Table Cell Properties dialog
- Table Detail Properties dialog
- Table Footer Properties dialog
- Table Header Properties dialog
- Table Properties dialog
- Table Wizard
- Tabular Cell Properties dialog
- Tabular Properties dialog
- To Chart dialog
- To Crosstab dialog
- Unique Key dialog
- Web Report Wizard
- Windows Media Properties dialog

Add Aggregation dialog

This dialog appears when you expand the Dynamic Resource > Aggregations node in the Resources panel, then click <Add Aggregation•gt; or right-click a dynamic aggregation and then select Edit from the shortcut menu. It helps you to create or edit a dynamic aggregation in a report.



Aggregation Name

Specifies the name of the aggregation.

Mapping Name

Click ... to select a field or a formula on which the aggregation is based on in the Select Resource dialog.

Aggregate Function

Specifies the function for the aggregation. The function will be applied to the group where the aggregation is placed or the entire report if the aggregation is placed in the report header or footer. For details about each function, see Math functions.

OK

Creates or edits the aggregation and closes the dialog.

Cancel

Cancels the operation and closes the dialog.

Displays the help document about this feature.

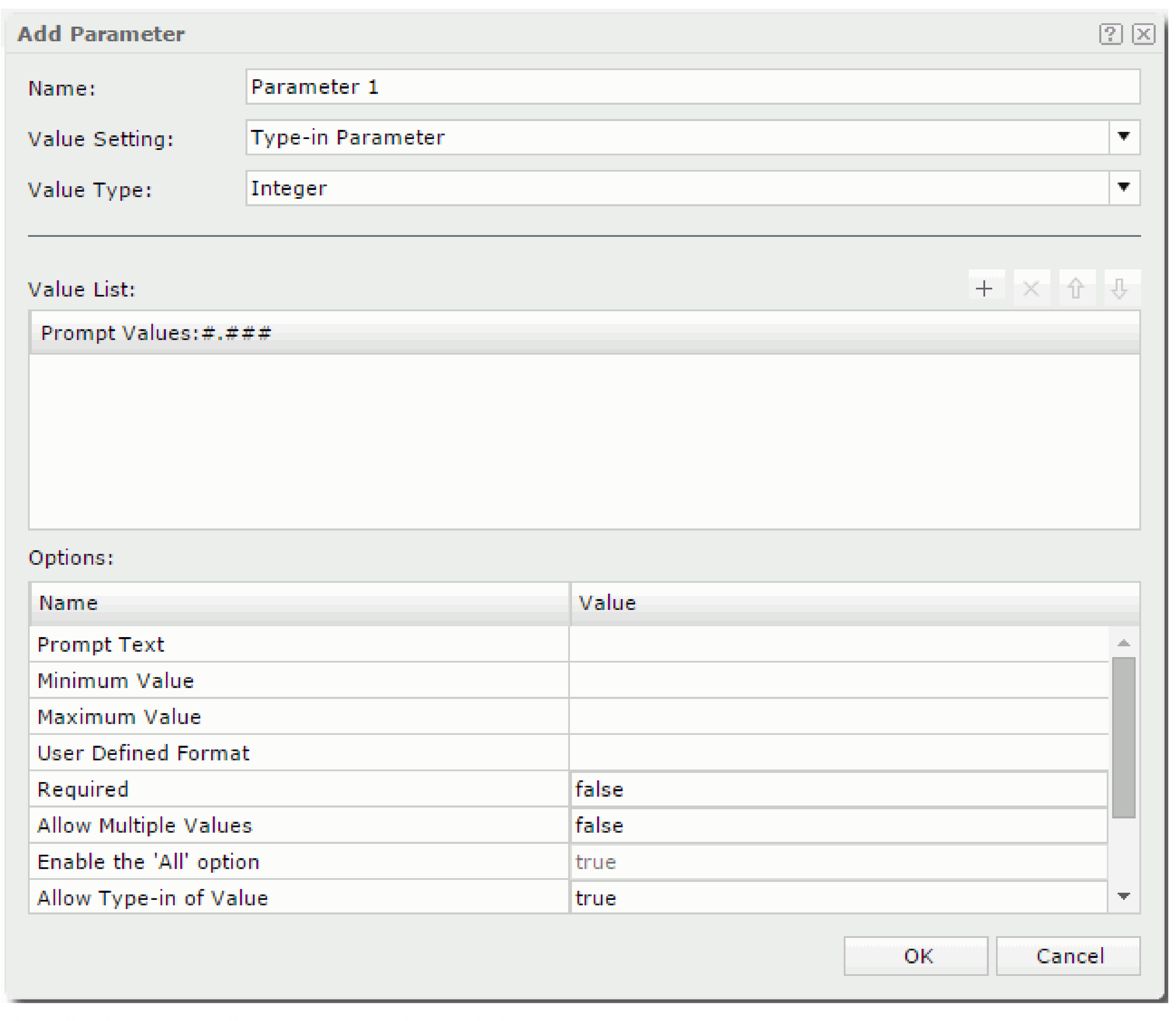
Ignores the setting and closes this dialog.

Related topics:

- Using dynamic resources

Add Parameter dialog

The dialog appears when you click the button [ab] in the Value text field of the Query Filter dialog or Filter dialog and select **<Add Parameter•gt;** from the **Local Parameters** node of the drop-down list. It helps you to add a local parameter to the report.



The following are details about options in the dialog:

Name

Specifies the name of the parameter.

Value Setting

Specifies the parameter type.

- **Type-in Parameter**
  If selected, you can predefine parameter values by typing manually.
- **Bind with Single Column**
  If selected, the parameter will be bound with a DBField. The values of the DBField will be retrieved as the parameter values at runtime.
- **Bind with Cascading Columns**
  If selected, you can create a group of cascading parameters so as to achieve the function of filtering parameters with parameters in a simple way.

Value Type

Specifies the data type of the parameter.

Values

The section after Value Type varies with the type you select from the Value Setting drop-down list.

- **For Type-in Parameter**
  - **Value List**
    Lists the predefined parameter values.
  - **Prompt Values** [format hint]
    Predefines the parameter values. The [format hint] shows what the predefined values should look like.

    There can be more than one prompt value. All the prompt values must be of the same type as specified in the Value Type field.
    - +
      Adds a new prompt value to the list. Double-click in the value line to edit the value. If the parameter is of Date, DateTime, or Time type, you can also click the calendar button to set a date and time value in the Calendar dialog.
    - ×
      Removes the selected prompt value.
    - ⇧
      Adjusts the display sequence of selected value by moving it up a step.
    - ⇩
      Adjusts the display sequence of selected value by moving it down a step.
- **For Bind with Single Column**
  JReport provides such a mechanism that links each field value of Display Column with the exact value of Bind Column. When you select the field value from Display Column, the field value of Bind Column is actually sent to the query and filters the query result. It helps you to bind the parameter to a DBField (column) in order to provide a list of DBField values for users to select at runtime. The selected value of the bound DBField will be used as the parameter value. Additionally, instead of displaying the values of the bound one, you can also choose to display values of other DBField, which probably makes more sense to the report end users. For example, it might be confusing if you provide a list of customer ID numbers for users to select at runtime, since the ID numbers could probably mean nothing to them. In cases like this, it is wise for you to display values of other fields, which could probably make more sense. For this case, you might prefer to display customer names instead of ID numbers. When the report end users select a customer name from the list, its ID number is passed to the query as the parameter value so that the search criteria can be fulfilled. So, before the report with a parameter bound to a DBField is processed, a query statement will be executed first to retrieve the values of the bound field and display field.
  - **Source**
    Specifies the business view in the current data source from which to retrieve the columns which can be used to bind with the parameter. Select one from the drop-down list.
  - **Bind Column**
    Specifies the DBField to be bound with the parameter.
  - **Display Column**
    When viewing a report which contains the parameter, the field values related to the column name you selected in the drop-down list of Display Column will be displayed in the Enter Parameter Values dialog at runtime.
- **For Bind with Cascading Columns**
  - **Source**
    Specifies the business view in the current data source from which to retrieve the columns which can be used to bind with the parameters. Select one from the drop-down list.
  - **Value List**
    Defines a group of cascading parameters.
    - +
      Adds a new parameter.
    - ×
      Removes the selected parameter.
    - ⇧
      Adjusts the parameter level by moving it up a step. The higher the position is, the higher the level is. The lower level field values are controlled by the higher level field values.
    - ⇩
      Adjusts the parameter level by moving it down a step.
    - **Bind Column**
      Specifies the DBField to be bound with the parameter.
    - **Display Column**
      When viewing a report which contains the parameter, the field values related to the column name you selected in the drop-down list of Display Column will be displayed in the Enter Parameter Values dialog at runtime.
    - **Parameter**
      Specifies whether to set a group of Bind Column and Display Column as a parameter which will become a member of the cascading parameter group. All parameters in the cascading group will be created as independent parameters in the catalog after you finish the dialog.

      The parameter name is provided by JReport automatically, and the format is *[The name specified as parameter name]-[Bind Column name]*.

Options

Specifies options for the parameter. The options vary with the type you select from the Value Setting drop-down list.

- **Prompt Text**
  Specifies the text for prompting users to provide the value of the parameter.

  Tips:
  - To help users avoid typing a value out of range, you can add the value range to the end of the prompt text. For example, "Type a customer ID (1~100):".
  - To help users avoid typing a value in a wrong pattern, you can add the value pattern to the end of the prompt text. For example, "Type an order date (MMM-dd-yy):".
- **Minimum Value/Length**
  Specifies the minimum value allowed for the parameter. If the parameter value type is String, this option is defined as Minimum Length, which means that you can set the minimum allowed length of the string.
- **Maximum Value/Length**
  Specifies the maximum value allowed for the parameter. If the parameter value type is String, this option is defined as Maximum Length, which means that you can set the maximum allowed length of the string.
- **String Encoding**
  Specifies the encoding for the parameter. Available only when the parameter's data type is defined as String.
- **User Defined Format**
  For certain types of parameters, the value format supplied by JReport may not satisfy your requirement. In this case, you can define your own preferred format. When you have defined the value format, all the values you supply at runtime should be based on it. Otherwise there will be an error message.

  Notes:
  - The letters that stand for the user defined format should follow the JDK standard. In addition, the user defined format only limits the format of the input value, which does not mean that the display value will be in the same format. To change the display format, you will need to specify the Format property of this parameter in the Report Inspector.
  - The user defined format will show different appearance under different locale, and you need input values according to the displayed format. For example, under the Spanish or French locale, ####.## will be displayed as #0,## and you should follow #0,## to input values.
- **Required**
  If set to true, the parameter will be treated as a required parameter. A required parameter must be fulfilled, which means that you must supply a value in the Enter Parameter Values dialog before the report can be run. If set to false, the parameter will be treated as an optional parameter. An optional parameter can be left blank, which means you can either type a value, or leave it as it is.

  In the Enter Parameters Value dialog, the prompt text of a required parameter will be marked with a * sign.

  Note: If the parameter is optional and you do not expect it to take effect in your report, in the Enter Parameter Values dialog, clear the content in the value field. Otherwise, any value left in the field will function in reports.
- **Hide Parameter When Single Value Returned**
  Specifies to hide the parameter at runtime when only a single value is returned for the parameter. The returned single value will be used as default value of the parameter.
- **Allow Multiple Values**
  Enables multiple parameter values to be selected. If it is set to true, when running a report with the parameter, the Enter Values dialog is provided for you to specify multiple values.

  Note: Multi-value parameters can only be supported in dataset filters, query filters, imported SQL files, and subreport/link report conditions.
- **Enable the "All" Option**
  Displays the All checkbox in the Enter Values dialog which means that the parameter value result is all the values that match the parameter data type. Available when Allow Multiple Values is set to true.
- **Treat Blank as Null**
  Available only when the parameter's data type is defined as String. If it is true, the parameter value will be null; if false, the default parameter value is "".
- **Use Current Date-Time**
  If set to true, the current date time in your system will be used as value of the parameter. Available only when the parameter's data type is defined as Date, Time or DateTime.
- **Use Current Date-Time When Blank**
  If set to true, when the value of the parameter is blank, the current date time in your system will be used as value of the parameter. Available only when the parameter's data type is defined as Date, Time or DateTime.

  If the type of the parameter is set to Date, Time or DateTime, you can choose whether to get parameter value dynamically from the system's date time by enabling one of the above options - Use Current or Use Current When Blank. However, if you want to use this feature, you need to make sure your system's DateTime format is set to yyyy-MM-dd H:mm:ss as it is the only format JReport supports; otherwise, you need to make use of the User Defined Parameter Format feature. Also, if one of the two options is enabled, you need to provide the parameter's prompt value according to the data type of your parameter.
  - When the data type of the parameter is Date, you can specify the value according to the format yyyy-MM-dd.
    - yyyy-01-01: Year will be dynamically changed.
    - 2007-MM-01: Month will be dynamically changed.
    - 2007-01-dd: Date will be dynamically changed.
    - yyyy-MM-01: Year and month will be dynamically changed.
    - yyyy-01-dd: Year and date will be dynamically changed.
    - yyyy-MM-dd: Year, month, and date will be dynamically changed.
    - 2007-MM-dd: Month and date will be dynamically changed.
  - When the data type of the parameter is Time, you can specify the value according to the format H:mm:ss.
    - H:00:00: Hour will be dynamically changed.
    - 12:mm:00: Minute will be dynamically changed.
    - 12:00:ss: Second will be dynamically changed.
    - H:mm:00: Hour and minute will be dynamically changed.
    - 12:mm:ss: Minute and second will be dynamically changed.
    - H:00:ss: Hour and second will be dynamically changed.
    - H:mm:ss: Hour, minute and second will be dynamically changed.
  - When the data type of the parameter is DateTime, specify the value in a combination of the above.
- **Sort**
  Specifies how to sort values of cascading parameters.
  - **Ascend**
    Sorts parameter values in an ascending order.
  - **Descend**
    Sorts parameter values in a descending order.
  - **No Sort**
    The parameter values are displayed in the original order as in database.
  - **Sort By**
    Sorts parameter values by sorting other fields in the same data source as the cascading group. When Sort By is selected, the Custom Sort dialog will be displayed for specifying fields and their sorting orders.
- **Distinct**
  If set to true, when viewing a report that contains the parameter, in the Enter Parameter Values dialog, identical values will appear only once and no duplicate values will be listed.
- **Allow Type-in of Value**
  Specifies whether to enable typing values besides selecting values when setting parameter values. The property is enabled for Type-in Parameter by default.
- **When Out of Value Range**
  Specifies how JReport will respond if the typed value is out of the predefined value list. The property is enabled only when the option Allow Type-in of Value is set to true.

  There are three solutions as follows:
  - **Adopt directly**
    The typed value will be used directly.
  - **Use default**
    The default value defined when creating the parameter will be used.
  - **Warn**
    Displays a warning message telling that "Value xxx is out of the available value list".

    Note: For a multi-value parameter, if one of the input values is not in the value list, the whole input result will be treated as out of range.
- **Get Value from API Only**
  Specifies whether the parameter values can be input via GUI or not.

  If true, the parameter values should only be obtained from API and there will be no parameter dialogs available on any JReport Designer or Server GUI for inputting values. Users will have to give the values of the parameter by API such as URLs, coding, and sessions, otherwise the default value will be used.

  If false, the parameter can be obtained from both GUI and API.

  If a member of a cascading parameter group is set to get values from API only, it means that all the higher level members shall get values from API only. This also applies to parameters that some other parameters depend on. If in JDashboard several parameters share values, when any of them is set to get values from API only, they will share the values from API.
- **On Parameter Value Change**
  Specifies to define an action that will be called when the parameter value changes. Choose a formula from the drop-down list to develop an action, which will be called before the other actions defined in the report or library component. It can work as a parameter value validation rule.

OK

Applies all changes and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

[?]

Displays the help document about this feature.

[X]

Ignores the setting and closes this dialog.

Related topics:

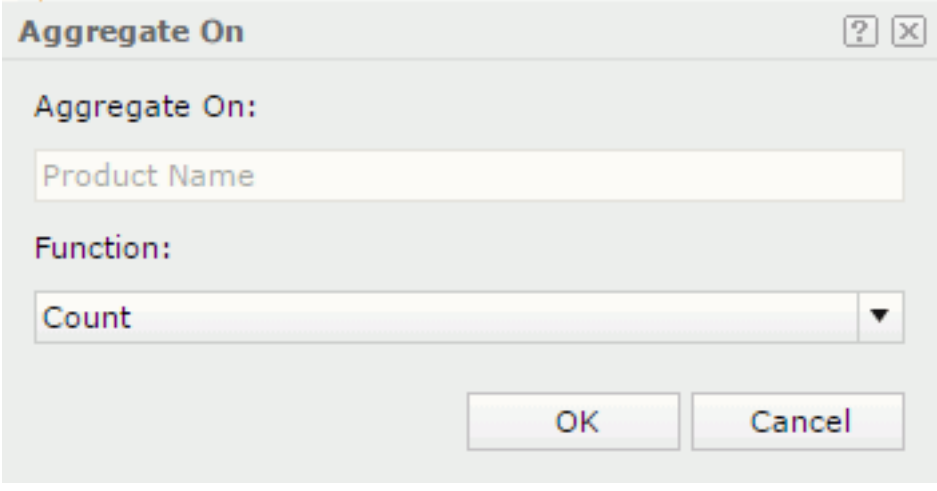
- Applying parameters

Aggregate On dialog

The dialog appears when you do either of the following:

- Right-click the field in a table detail column and select Aggregate On from the shortcut menu.
- Click on a column header to select the column, then on the Context toolbar, click the Aggregate On button Σ.

It helps you to summarize data in the selected table detail column.



Aggregate On

Displays the field bound with the detail column. The field is also the one on which the summary is based.

Function

Specifies the function to summarize the field in the detail column. For details about each function, see Math functions.

OK

Summarizes the detail column data with the specified function and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Displays the help document about this feature.

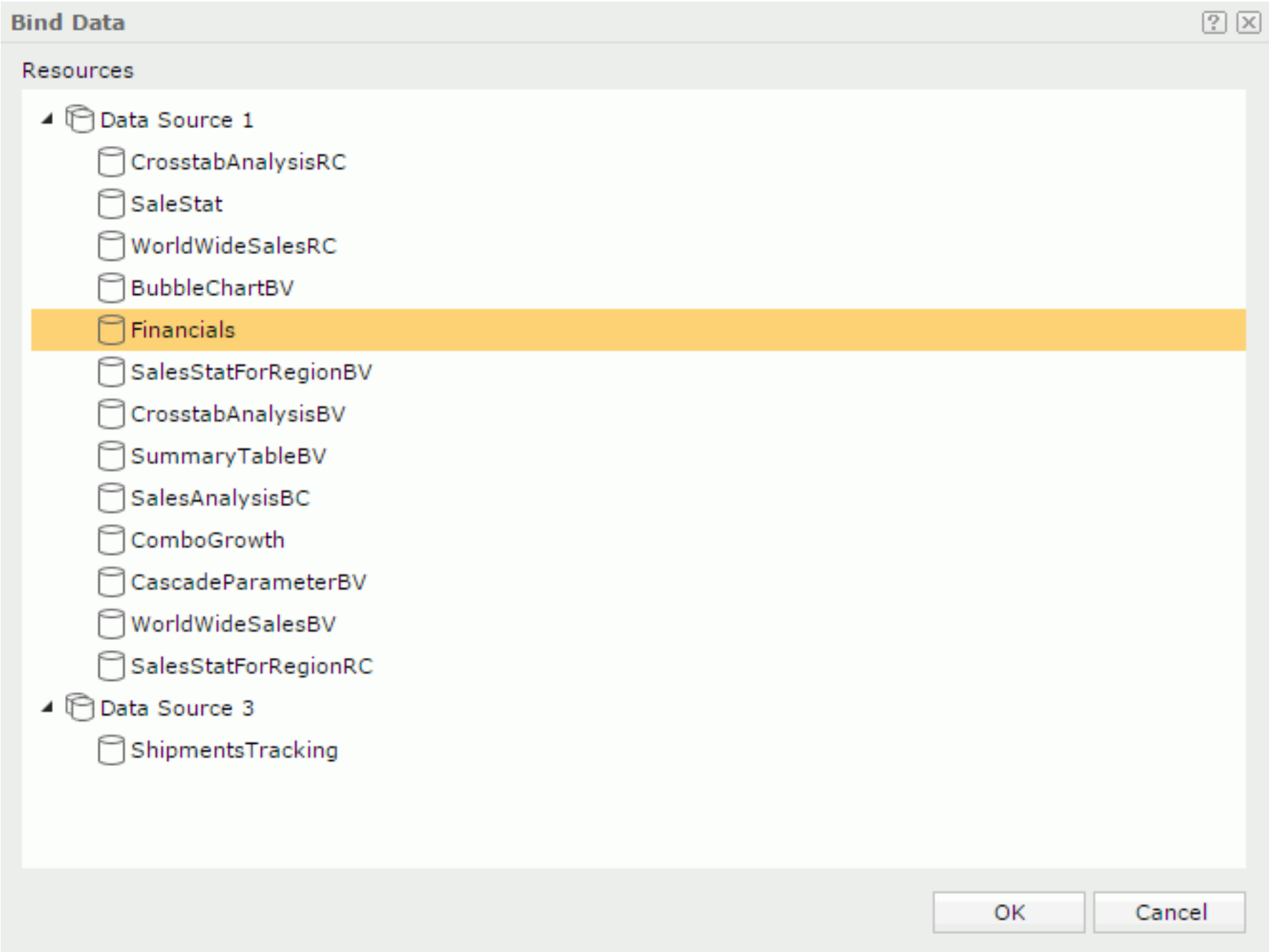
Ignores the setting and closes this dialog.

Related topics:

- Aggregating on a detail column

Bind Data dialog

This dialog appears when you click Menu > Edit > Bind Data. If you want to use formulas to control whether to hide or show objects whose parents don't have data source in a web report, you first need to bind a data source to the web report, then you can create formulas in the data source to control the Invisible property of such objects in the web report in the Show Objects dialog.



Available data resources

Lists all the business views in the current catalog that can be used as data source of the web report. Select the one you want to use.

OK

Binds the data to the web report.

Cancel

Does not retain the changes and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Related topics:

- Using formulas to control showing or hiding components

Button Properties dialog

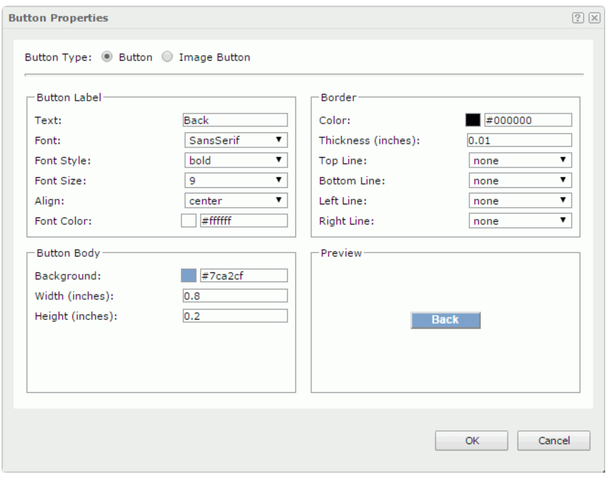
This dialog appears when you right-click a button in a navigation control and select Properties from the shortcut menu. It helps you to modify the properties of the button.

Button Type

- Button
  The button is displayed as a normal button.
- Image Button
  The button is displayed as an image.

Button

When Button is selected as the button type, the following options are available.



Button Label

Specifies the properties of the button label.

- **Text**
  Specifies the text of the label.
- **Font**
  Specifies the font face of the text.
- **Font Style**
  Specifies the font style of the text.
- **Font Size**
  Specifies the font size of the text.
- **Align**
  Specifies the alignment way of the label in the button.
- **Font Color**
  Specifies the font color of the text.

Border

Specifies the properties of the button border.

- **Color**
  Specifies the border color.
- **Thickness**
  Specifies the border width.
- **Top Line**
  Specifies the style of the top border line.
- **Bottom Line**
  Specifies the style of the bottom border line.
- **Left Line**
  Specifies the style of the left border line.
- **Right Line**
  Specifies the style of the right border line.

Button Body

Specifies the properties of the button body.

- **Background**
  Specifies the background color of the button body.

  To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.
- **Width**
  Specifies the width of the button.
- **Height**
  Specifies the height of the button.

Preview

Displays a preview of the button according to the button properties.

Image Button

When Button Image is selected as the button type, the following options are available.

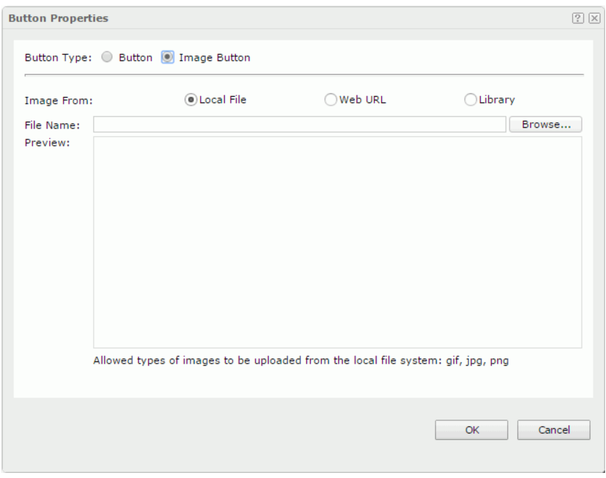


Image From

Specifies the source of the image file.

- **Local File**
  Specifies to use an image from the local file system.
  - **File Name**
    Specifies the path and name of the image file. You can click the Browse button to locate the image file.
- **Web URL**
  Specifies to use an image via URL.
  - **Image URL**
    Specifies the URL of the image file. JReport will record the latest 10 entered URLs in the drop-down list.

    Note: If your JReport Server is in an intranet, to successfully access the image via URL, you need to add the parameters `-Dhttp.proxyHost=XXX -Dhttp.proxyPort=XX` to the server's startup file JRServer.bat, which locates in `<install_root>\bin`.
- **Library**
  Specifies to use an existing image.
  - **My Images**
    The My Images folder is a virtual location where JReport Server stores the images that have once been inserted into reports. Select the one you want to use.

Preview

Displays a preview of the selected image.

OK

Applies the button properties and closes this dialog.

Cancel

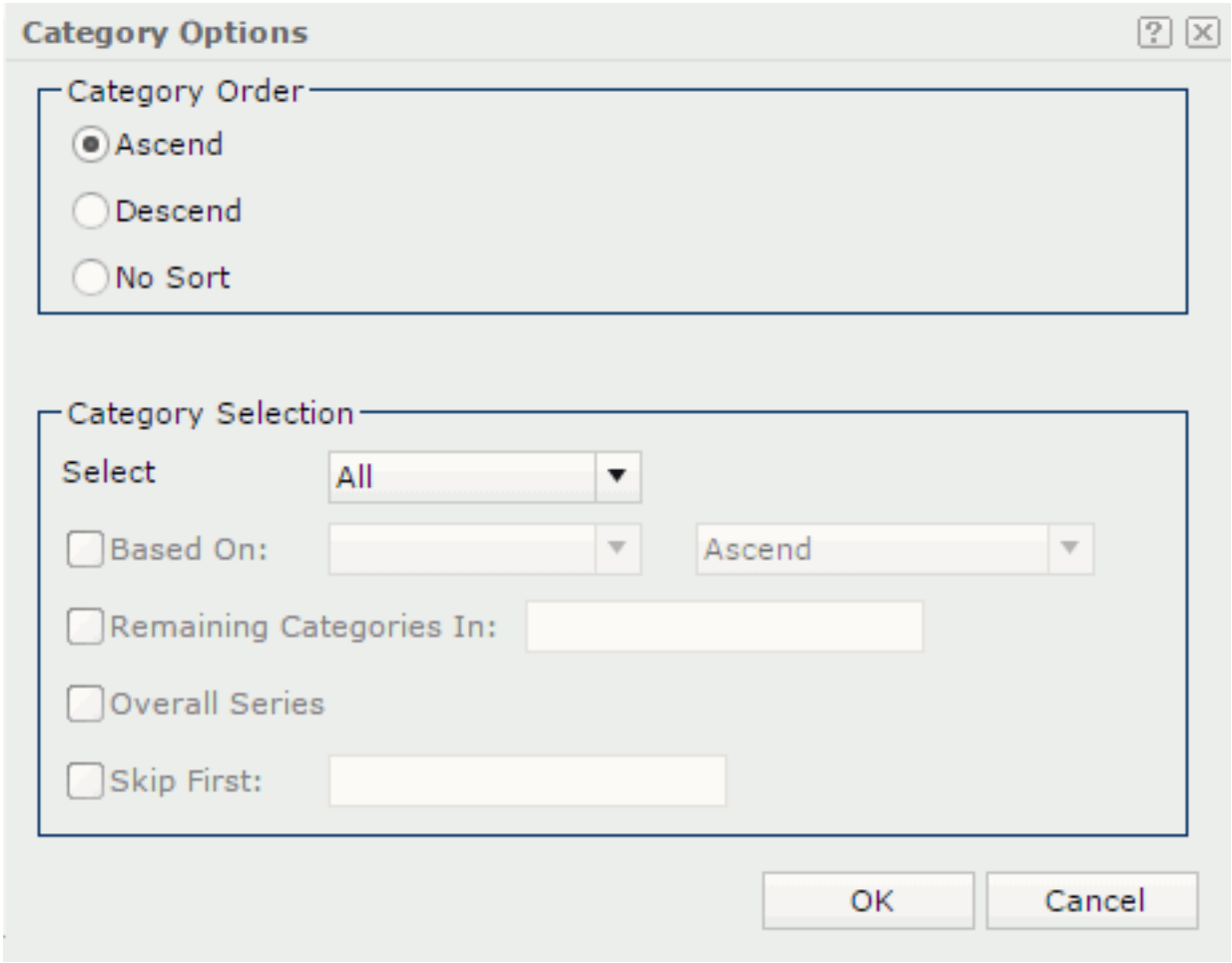
Cancels the changes and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Category Options dialog

This dialog appears when you click the Top N button above the Category box in the Bind Data screen of chart in the Web Report Wizard, Insert Chart dialog, Chart Wizard, or To Chart dialog. It helps you to set the sort order of the category values and define the number of the category values that will be displayed in the chart.



Category Order

Specifies in which order data on the category axis of a chart will be displayed.

- **Ascend**
  Lists data in an ascending order.
- **Descend**
  Lists data in a descending order.
- **No Sort**
  Keeps the data in their original order in database.

Category Selection

Specifies the number of the category values that will be displayed in the chart.

- **Select**
  Specifies the Select N condition to define the number of the category values that will be displayed.
  - **All**
    If selected, all category values will be displayed.
  - **Top N**
    If selected, specify a number in the field to the right and the first N category values will be displayed.
  - **Bottom N**
    If selected, specify a number in the field to the right and the last N category values will be displayed.
- **Based On**
  If checked, the category values will be sorted by values of the summary selected with the direction specified. If unchecked, the category values will be sorted by the order specified in the Category Order box of the dialog.
- **Remaining Categories In**
  Enabled when Top N or Bottom N is selected from the Select drop-down list. Check this option and then type a character string in the text field to group all the category values beyond the top/bottom N range.
- **Overall Series**
  Specifies whether to calculate the top or bottom N category values based on the series values. Enabled when Top N or Bottom N is selected from the Select drop-down list.
- **Skip First**
  If you check the Skip First option and input a number M in the text field to the right, then the first M category values in the chart will be skipped and the Select N condition will take effect beginning with M+1. The skipped values will be included in the Remaining Categories group together with all the category values beyond the top/bottom N range.

OK

Accepts the changes and closes the dialog.

Cancel

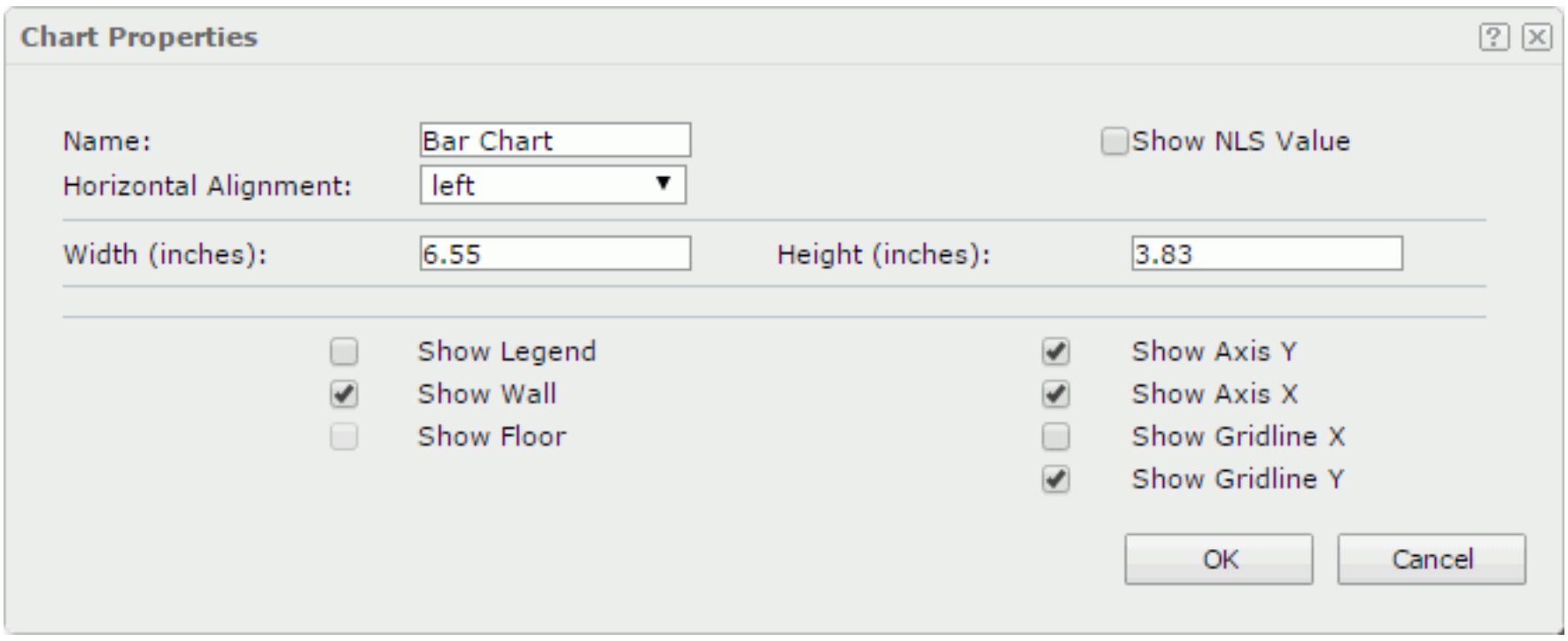
Discards the changes and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Chart Properties dialog

The dialog appears when you right-click a chart and select Properties from the shortcut menu. It helps you to specify the properties of the chart.



Name

Specifies the display name of the chart, which will be shown on the shortcut menu of the chart.

Show NLS Value

Specifies to show the translated name of the display name of the chart in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the chart is not modified.

Horizontal Alignment

Specifies the horizontal justification of the chart. Choose an option from the drop-down list.

- **left**: Aligns the chart on the left of the report.
- **right**: Aligns the chart on the right of the report.
- **center**: Aligns the chart in the center of the report.

Width

Specifies the width of the chart.

Height

Specifies the height of the chart.

Show Legend

Specifies whether to make the legend in the chart visible.

Show Wall

Specifies whether to show the wall in the chart.

Show Floor

Specifies whether to show the floor, only for 3-D chart types.

Show Axis Y

Specifies whether to show the axis Y.

Show Axis X

Specifies whether to show the axis X.

Show Gridline X

Specifies whether to show the gridlines perpendicular to the axis X.

Show Gridline Y

Specifies whether to show the gridlines perpendicular to the axis Y.

OK

Applies the chart properties and closes this dialog.

Cancel

Cancels the changes and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Chart Wizard

This wizard appears when you do one of the following:

- Select a chart, then click Menu > Edit > Wizard.
- Select a chart, then click the Chart Wizard button on the Context toolbar.
- Right-click the icon of a chart or any part of a chart other than the legend and label, then select Chart Wizard from the shortcut menu.

It helps you to edit a chart and varies with different chart types: common chart types, organization chart, heat map.

OK

Applies the changes and closes the wizard.

Cancel

Cancels the changes and closes the wizard.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

For common chart types

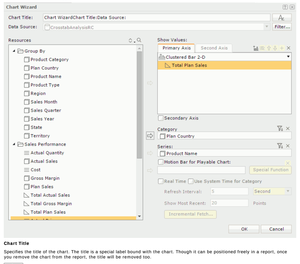


Chart Title

Specifies the title of the chart. The title is a special label bound with the chart. Though it can be positioned freely in a report, once you remove the chart from the report, the title will be removed too.

Specifies the font properties of the chart title.

- **Font**
  Lists all the available font faces that can be selected to apply to the title.
- **Font Style**
  Specifies the font style of the title. It can be selected from the following: plain, bold, italic, and bold italic.
- **Size**
  Specifies the font size of the title.
- **Align**
  Specifies the position of the title to be left, right, center or justify.
- **Font Color**
  Specifies the font color of the title.
- **Background Color**
  Specifies the background color of the title.

Data Source

Displays the business view used by the chart.

Filter

Opens the Query Filter dialog to specify the filter which you want to apply to the selected business view.

Resources

Displays the resources that can be added to the chart.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on .Report Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.

Lists more search options.

- **Highlight All**
  Specifies whether to highlight all matched text.
- **Match Case**
  Specifies whether to search for text that meets the case of the typed text.
- **Match Whole Word**
  Specifies whether to search for text that looks the same as the typed text.

Highlights the previous matched text.

Highlights the next matched text.

Adds the selected resource to be displayed in the chart.

Show Values

Lists the values you want to show in the chart. For a real time chart, the values you add must be of Numeric type and cannot be aggregation objects.

- **Primary Axis**
  Adds a chart type to the primary axis.
- **Secondary Axis**
  Adds a chart type to the secondary axis. Active only when the option Secondary Axis is checked.
- **X Axis**
  Lists the value you want to show on the X axis of the bubble chart.
- **Y Axis**
  Lists the value you want to show on the Y axis of the bubble chart.
- **Radius**
  Lists the value you want to show as the bubble radius.

Adds a combo chart to the Primary Axis or Secondary Axis.

Opens the Edit Additional Value dialog to edit the selected additional value.

Moves the selected view element one level up.

Moves the selected view element one level down.

Adds a new pair of Y Axis and Radius for the bubble chart.

Secondary Axis

Specifies whether to show the secondary axis in the chart.

Category

Lists the group object that will be displayed on the category axis of the chart.

For a real time chart, if no object is specified on the category axis, Use System Refresh Time will be automatically displayed in the Category text box, namely, the time at which the chart refreshes itself will be used as the category value.

Series

Lists the group object that will be displayed on the series axis of the chart. Not available in real time chart.

Opens the Category Options dialog or Series Options dialog to define the sort order of the category or series values and specify the number of the category or series values that will be displayed in the chart.

Motion Bar for Playable Chart

Lists the group object you want to use as the motion field. A motion field can only be of Integer, Date or Time data type. Available in maple bar, bench and bubble chart types only.

- **Special Function**
  Specifies only when the motion field is of Date data type. Click it to define the special function.
  - **Field**
    Displays on which field the special function will be applied.
  - **Function**
    Specifies the special function to the field.
  - **OK**
    Accepts the special function settings and leaves the dialog.
  - **Cancel**
    Cancels the special function settings and leaves the dialog.

Real Time

Specifies to run the chart in real time mode, which means it will be updated automatically by using real time data. Available to single bar, bench, line, and area chart types only.

- **Use System Time for Category**
  Specifies to use the time at which the chart refreshes itself as the category value.
- **Refresh Interval**
  Specifies the time interval at which the chart will get data and refresh itself automatically.
- **Show Most Recent N Points**
  Specifies the number of records that will be kept for the real time data on the chart.
- **Incremental Fetch**
  Opens the Unique Key dialog to configure a unique key for the real time chart.

Removes the selected resource.

For organization chart

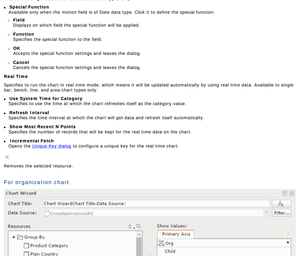


Chart Title

Specifies the title of the chart. The title is a special label bound with the chart. Though it can be positioned freely in a report, once you remove the chart from the report, the title will be removed too.

Specifies the font properties of the chart title.

Data Source

Displays the business view that has been used in the chart.

Filter

Opens the Query Filter dialog to specify the filter you want to apply to the selected business view.

Resources

Displays the resources that can be added to the chart.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

Launches the quick search toolbar to search for view elements.

Show Values

- **Primary Axis**
  Select Org from the chart type drop-down menu.
- **Child**
  Adds a field from the Resources box which identifies the entity by selecting both the field and Child and then clicking
- **Parent**
  Adds a field from the Resources box which shows the "reporting to" relationship among the entity members, that is, which child field member reports to or belongs to which child field member, by selecting both the field and Parent and then clicking

Removes the selected child or parent field.

Properties

The properties box presents a node model of the org chart. Data fields, labels and images can be inserted into the node as the information about the entity in the org chart, by using . By default all added objects are placed at the left top of the node, you need to adjust their positions and sizes in the node. You can also resize the node if required.

To remove an object from the node, select it and then click .

For heat map

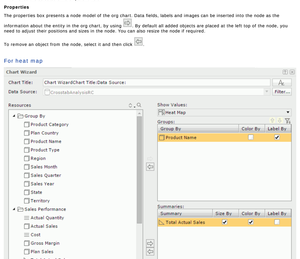


Chart Title

Specifies the title of the chart. The title is a special label bound with the chart. Though it can be positioned freely in a report, once you remove the chart from the report, the title will be removed too.

Specifies the font properties of the chart title.

Data Source

Displays the business view that has been used in the chart.

Filter

Opens the Query Filter dialog to specify the filter you want to apply to the selected business view.

Resources

Displays the resources that can be added to the chart.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

Launches the quick search toolbar to search for view elements.

Adds the selected field into the Groups or Summaries box.

Removes the selected field from the Groups or Summaries box.

Show Values

Displays Heat Map as the selected chart type.

Groups

Lists the fields used to group the data. There should be at least one group.

- **Color by**
  Specifies whether to color by a group. 0-n groups can be used as the color-by fields.
- **Label by**
  Specifies whether to show a group field in the innermost rectangle. 0-n groups can be used as the label-by fields.

Moves the selected group field one level up.

Moves the selected group field one level down.

Opens the Group Options dialog to define the sort order of the group values and specify the number of the group values that will be displayed in the chart.

Removes the selected group field.

Summaries

Lists the summaries used as size-by/color-by or displayed in the innermost rectangle.

The summaries should match the groups. For example, if the groups level is A > B > C, the static summaries grouped by C can be inserted into the Summaries box, but the static summaries grouped by A, B or other fields cannot.

If no group is specified in the Groups box, you can insert any static summary. And its group-by field will be inserted into the Groups box automatically.

- **Size by**
  Specifies to size by one summary or none.
- **Color by**
  Specifies to color by one summary or none.
- **Label by**
  Specifies whether to show a summary in the innermost rectangle. 0-n summaries can be used as the label-by fields.

Color Picker dialog

This dialog helps you to specify a color for an object. It contains the following tabs:

- Standard tab
- Custom tab

OK

Closes this dialog and returns the color value specified.

Cancel

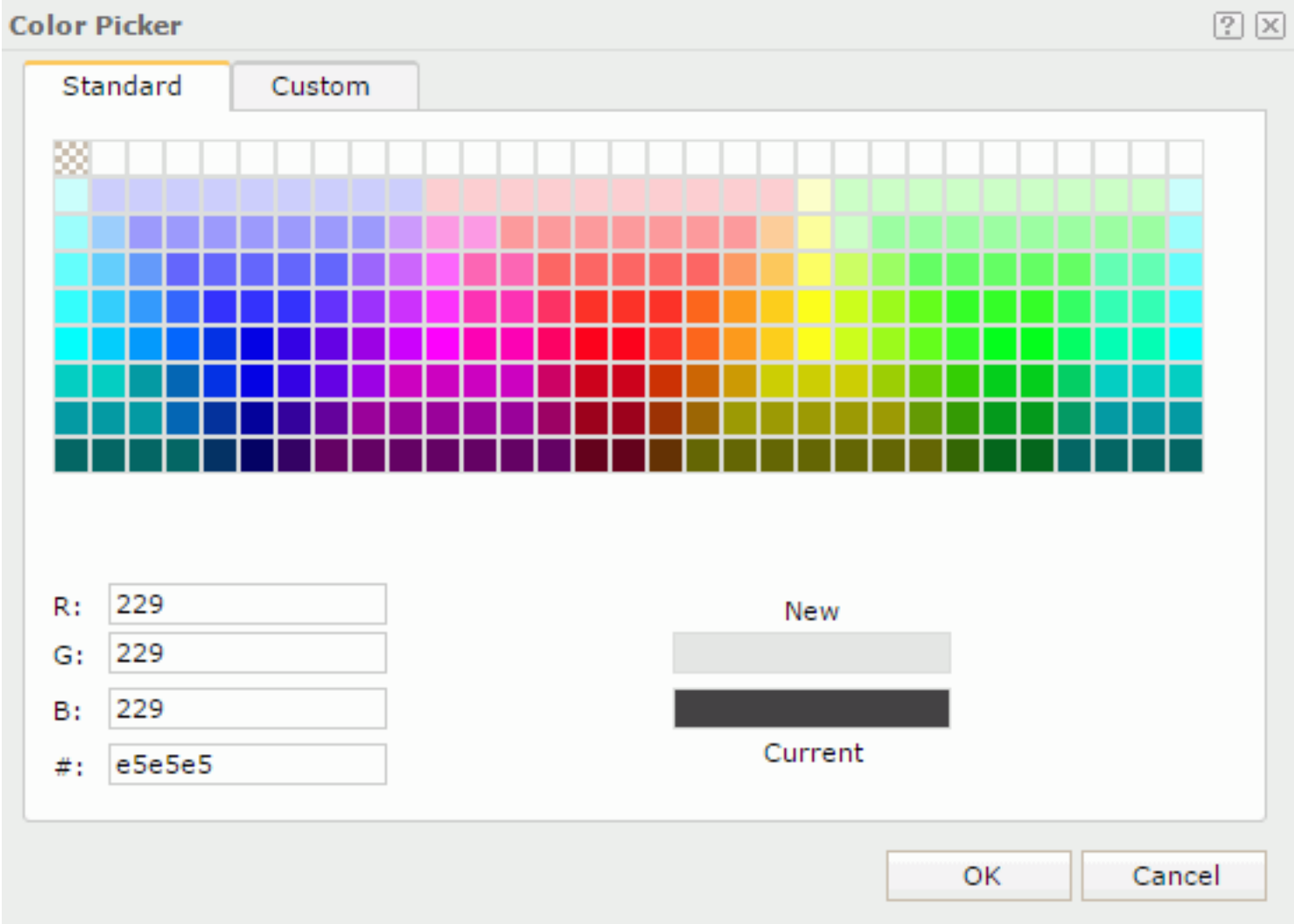
Closes this dialog without changing the color value.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Standard tab

This tab allows you to select one from the web safe color swatches.



Web Safe Colors

Lists the web safe colors for you to select one.

R

Specifies the amount of red in a color. The value is from 0 to 255.

G

Specifies the amount of green in a color. The value is from 0 to 255.

B

Specifies the amount of blue in a color. The value is from 0 to 255.

#

Specifies the hexadecimal value of the color.

New

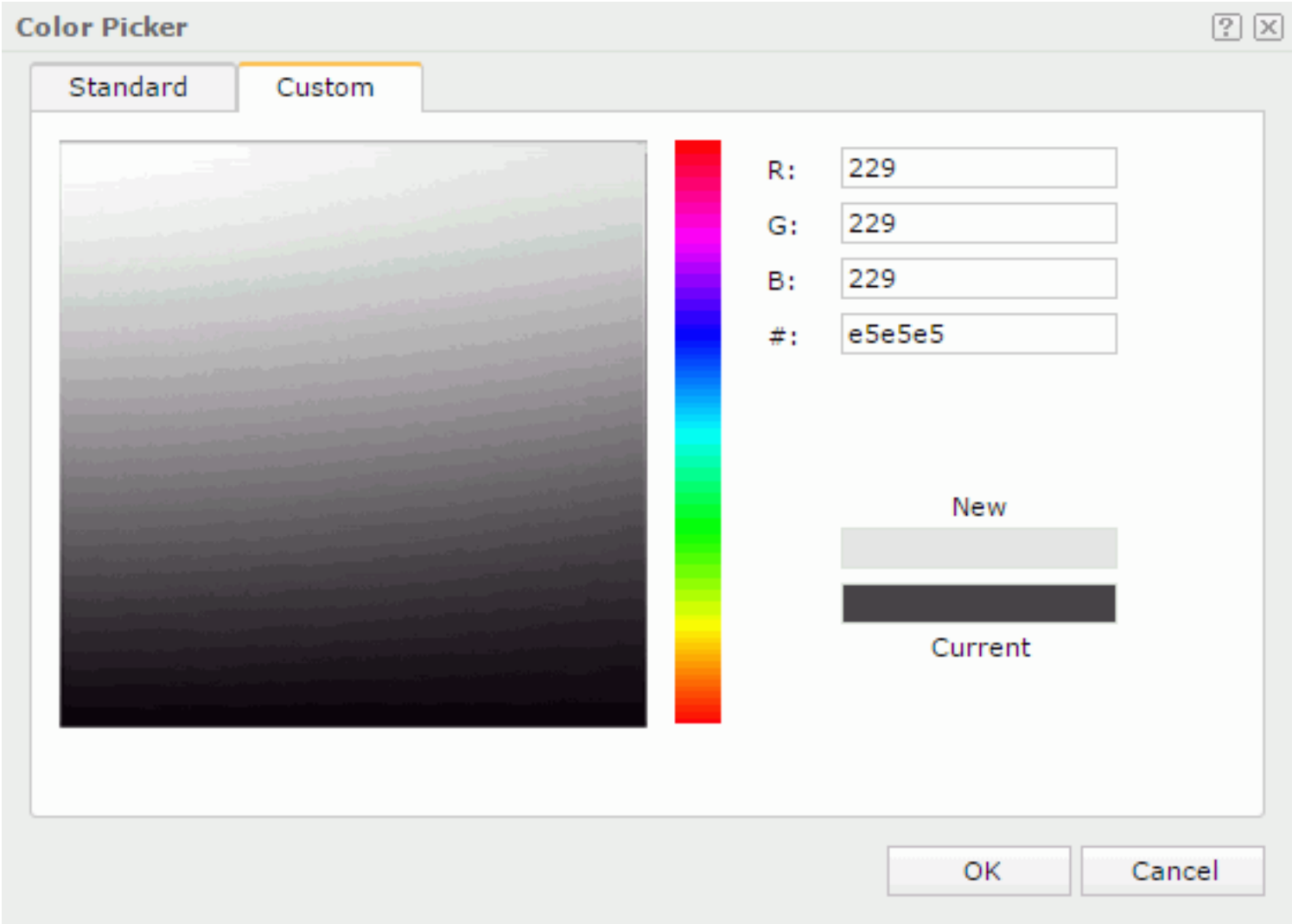
This part displays the color you select.

Current

This part always displays the original color. Clicking this part will restore the color to the original.

Custom tab

This tab allows you to customize the color within a wider range.



Color matrix

Select a color you want in the color matrix. The hue of the matrix is based on the color you define on the color bar.

Color bar

Specifies a color on the bar which will be the hue of the color matrix.

R

Specifies the amount of red in a color. The value is from 0 to 255.

G

Specifies the amount of green in a color. The value is from 0 to 255.

B

Specifies the amount of blue in a color. The value is from 0 to 255.

#

Specifies the hexadecimal value of the color.

New

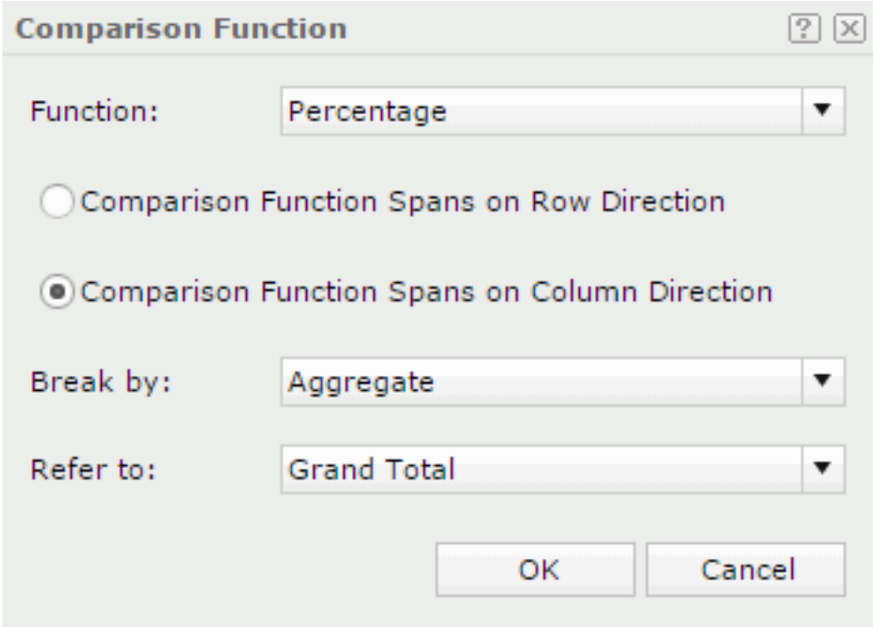
This part displays the color you select.

Current

This part always displays the original color. Clicking this part will restore the color to the original.

Comparison Function dialog

This dialog appears when you click the Comparison Function button in the the Insert Crosstab dialog or Crosstab Wizard, or in the Bind Data screen for crosstab in the Web Report Wizard. It helps you to add a comparison function as an aggregate field for a crosstab.



The following are details about options in the dialog:

Function

Specifies the function of the calculation: Percentage, Permillage, or Difference.

Comparison Function Spans on Row Direction

If checked, the comparison function will be placed into the column total cell of the crosstab.

Comparison Function Spans on Column Direction

If checked, the comparison function will be placed into the row total cell of the crosstab.

Break by

Items in the drop-down list vary with the position of the comparison function. It specifies the first parameter of the comparison function.

Refer to

Specifies the item as the other parameter of the comparison function.

OK

Accepts all changes and closes the dialog.

Cancel

Does not retain any changes and closes this dialog.

Displays the help document about this feature.

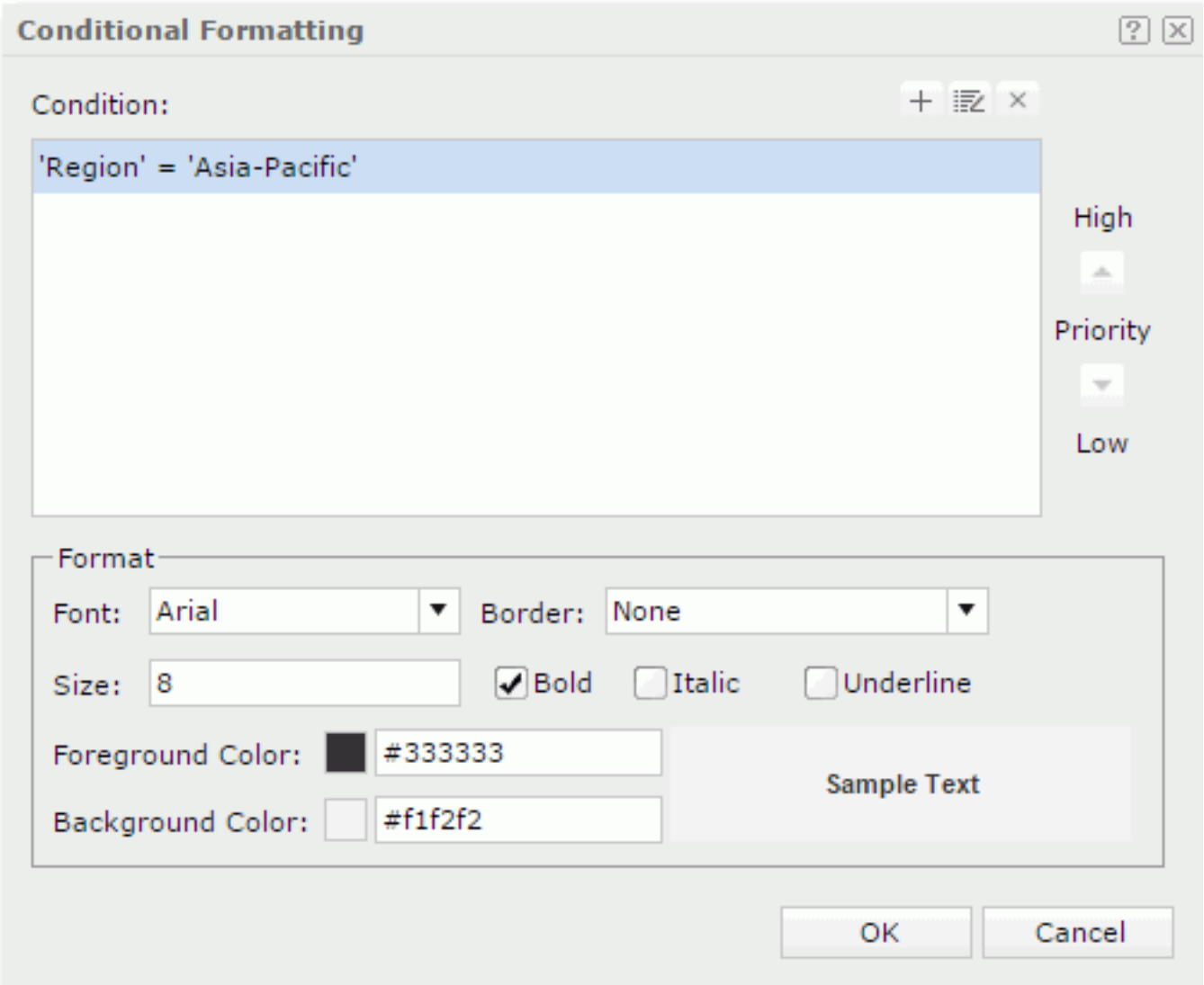
Ignores the setting and closes this dialog.

Related topics:

- Specifying comparison functions in a crosstab

Conditional Formatting dialog

This dialog appears when you right-click a field and select Conditional Formatting from the shortcut menu. It helps you to add some conditional formats to values of the selected field.



Condition

Displays all the conditions you have already added.

- Adds a new condition in the Edit Conditions dialog.
- Edits the selected condition.
- Removes the selected condition.
- **Priority**
  Specifies the priority of each condition.
  - Moves a condition up for a higher priority.
  - Moves a condition down for a lower priority.

Format

Specifies the format which will be applied to the field values when the specified condition is fulfilled.

- **Font**
  Specifies the font type for the field values.
- **Border**
  Specifies the border line style for the field values.
- **Size**
  Specifies the font size for the field values.
- **Bold**
  Specifies whether or not to bold the field values.
- **Italic**
  Specifies whether or not to make the field values italic.
- **Underline**
  Specifies whether or not to underline the field values.
- **Foreground Color**
  Specifies the foreground color for the field values.
- **Background Color**
  Specifies the background color for the field values.
- **Sample Text**
  Displays a preview sample of your settings.

OK

Applies the settings and exits the dialog.

Cancel

Cancels the settings and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Related topics:

- Adding conditional formats to fields

Crosstab Properties dialog

The dialog appears when you right-click a crosstab and select Properties from the shortcut menu. It helps you to specify the properties of the crosstab and contains the following tabs:

- General tab
- Border tab
- Crosstab tab
- Others tab

OK

Applies the settings and closes this dialog.

Cancel

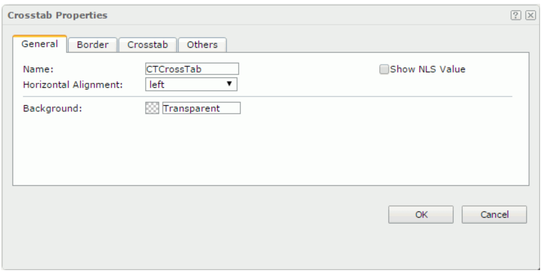
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the crosstab.



Name

Specifies the display name of the crosstab, which will be shown on the shortcut menu of the crosstab.

Show NLS Value

Specifies to show the translated name of the display name of the crosstab in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the crosstab is not modified.

Horizontal Alignment

Specifies the horizontal justification of the crosstab. Choose an option from the drop-down list.

- **Left**: Aligns the crosstab on the left of the tabular cell.
- **Right**: Aligns the crosstab on the right of the tabular cell.
- **Center**: Aligns the crosstab in the center of the tabular cell.

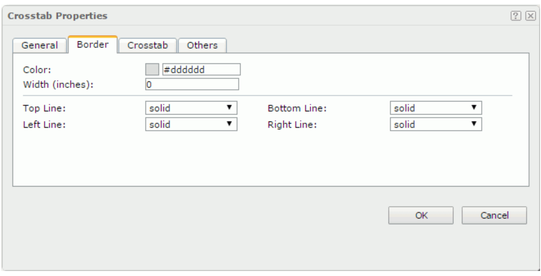
Background

Specifies the background color of the crosstab.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the crosstab.



Color

Specifies the color of the borders.

Width

Specifies the width of the borders.

Top Line

Specifies the line style of the top border. Choose a style from the drop-down list.

Bottom Line

Specifies the line style of the bottom border. Choose a style from the drop-down list.

Left Line

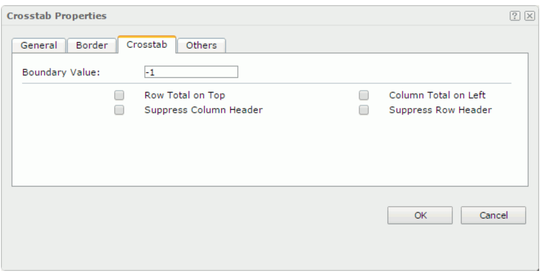
Specifies the line style of the left border. Choose a style from the drop-down list.

Right Line

Specifies the line style of the right border. Choose a style from the drop-down list.

Crosstab tab

This tab shows the layout-related information of the crosstab.



Boundary Value

Specifies the number of columns in one aggregate cell when the crosstab is displayed horizontally, or rows when displayed vertically.

Row Total on Top

Specifies whether or not to display the Total row in the first row of the crosstab.

Column Total on Left

Specifies whether or not to display the Total column in the first column in the crosstab.

Suppress Column Header

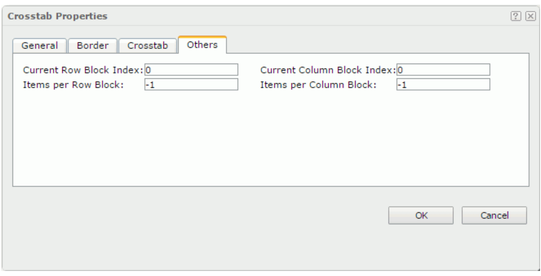
Specifies whether or not to suppress the column headers.

Suppress Row Header

Specifies whether or not to suppress the row headers.

Others tab

You can use this tab to view and configure some miscellaneous settings.



Current Row Block Index

Specifies the vertical index of the data block that will be displayed. 0 means the first block index, and 1 the second, and so on.

Current Column Block Index

Specifies the horizontal index of the data block that will be displayed. 0 means the first block index, and 1 the second, and so on.

Items per Row Block

Specifies the number of rows of records in each data block.

Items per Column Block

Specifies the number of columns of records in each data block.

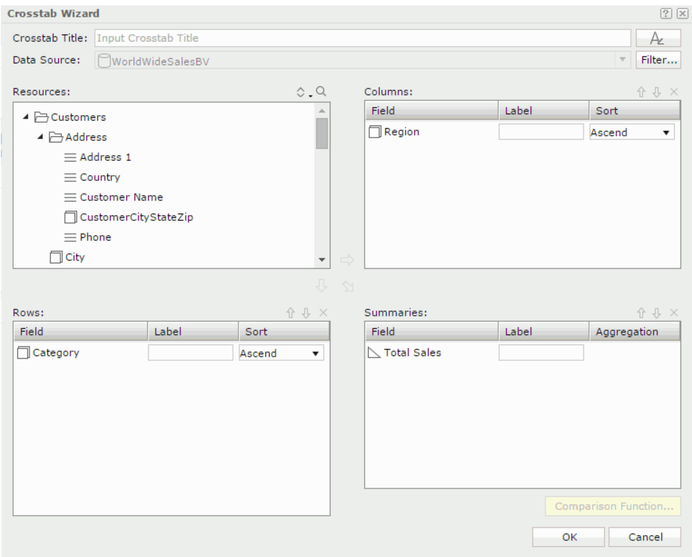
The four properties work together to control the data of the crosstab to be displayed in continuous page mode: Current Row Block Index, Current Column Block Index, Items per Row Block, and Items per Column Block.

Crosstab Wizard

This wizard appears when you do one of the following:

- Select a crosstab, then click Menu > Edit > Wizard.
- Select a crosstab, then click the Crosstab Wizard button on the Context toolbar.
- Right-click the icon of a crosstab and select Crosstab Wizard from the shortcut menu.

It helps you to change data of the crosstab.



Crosstab Title

Specifies a title for the crosstab.

Specifies the font properties of the crosstab title.

- **Font**
  Lists all the available font faces that can be selected to apply to the title.
- **Font Style**
  Specifies the font style of the title. It can be one of the following: plain, bold, italic, and bold italic.
- **Size**
  Specifies the font size of the title.
- **Align**
  Specifies the position of the title to be left, right, center or justify.
- **Font Color**
  Specifies the font color of the title.
- **Background Color**
  Specifies the background color of the title.

Data Source

Displays the business view used by the crosstab.

Filter

Opens the Query Filter dialog to specify the filter which you want to apply to the selected business view.

Resources

Displays the view elements in the selected business view.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.
-
  Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
-
  Highlights the previous matched text.
-
  Highlights the next matched text.

Adds the selected group object to be displayed on the columns of the crosstab.

Adds the selected group object to be displayed on the rows of the crosstab.

Adds the selected aggregation object or detail object to be the summary field of the crosstab.

Columns/Rows

- **Field**
  Lists the group objects that will be displayed in the columns/rows of the crosstab.
- **Label**
  Specifies the display names of the group objects. By default these are blank and no names will be created for the group objects to label the columns/rows. You can click in the cells to edit them if required.
- **Sort**
  Specifies the sort order of the group objects.

Summaries

- **Field**
  Lists the aggregation/detail objects that you select to create summaries.
- **Label**
  Specifies the display names of the aggregation/detail objects. By default these are blank and no names will be created for the objects to label the summaries. You can click in the cells to edit them if required.
- **Aggregation**
  Specifies the functions used to summarize data of the detail objects.
- **Comparison Function**
  Opens the Comparison Function dialog to add a comparison function as an aggregate for the crosstab.

Moves the selected view element one level up.

Moves the selected view element one level down.

Removes the selected resource.

OK

Applies the changes and closes the wizard.

Cancel

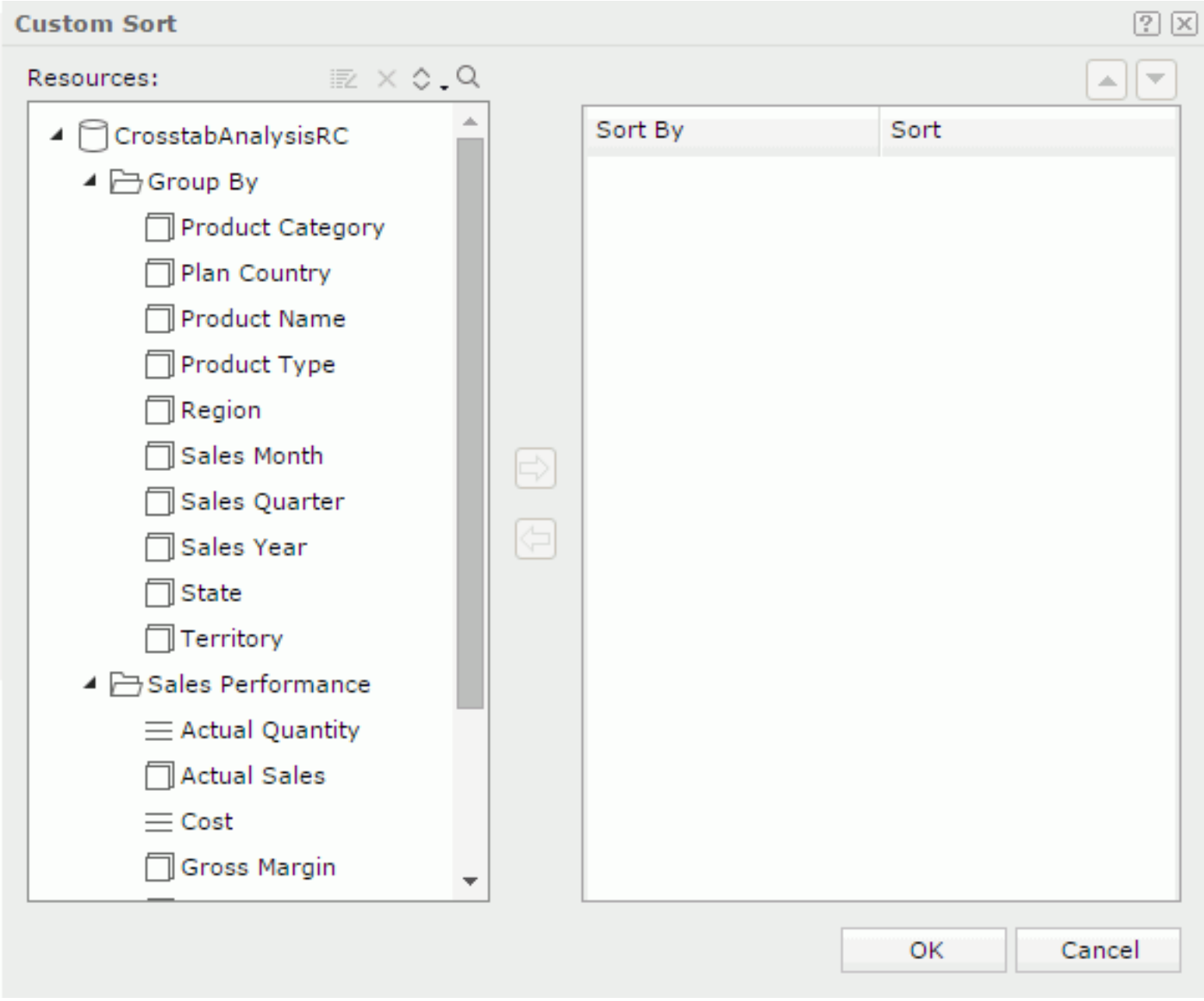
Does not retain changes and closes the wizard.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Custom Sort dialog

The dialog appears when you click the Sort Fields By button in the Display tab of the Insert Table dialog or Table Wizard, or in the Display tab for table in the Web Report Wizard, or select Custom Sort from the Sort column in the Group tab of the Insert Table dialog or Table Wizard, or in the Group tab for table in the Web Report Wizard. It helps you to set the sorting manner of data displayed in the table or the sorting manner of groups at the specified group level.



Resources

Lists all the available data resources.

Edits the selected dynamic resource. Available only when this dialog is opened from the web report wizard.

Removes the selected dynamic resource. Available only when this dialog is opened from the web report wizard.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.
-
  Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
-
  Highlights the previous matched text.
-
  Highlights the next matched text.

Sort By

Displays the fields on which the sorting will be based.

Sort

Specifies the sorting manner.

- **Ascend**
  If selected, data or groups in the table will be sorted based on values of the specified sort by field in ascending order.
- **Descend**
  If selected, data or groups in the table will be sorted based on values of the specified sort by field in descending order.

Adds the selected field as the sort by field.

Removes the specified sort by field if it is not required.

Moves the selected sort by field one step up.

Moves the selected sort by field one step down.

OK

Applies the settings and closes this dialog.

Cancel

Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Related topics:

- Sorting data in a table

Data Format dialog

This dialog appears when you click ... to specify the data format for a chart in the Inspector panel. The options in the dialog varies according to different sources it is opened from.

When you open the dialog to specify the data format for the category axis (the X axis) in a chart, it consists of two tabs: major label tab and minor label tab.

OK

Applies the settings and closes the dialog.

Cancel

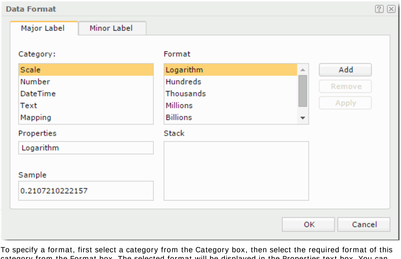
Cancels the settings and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Major Label tab

Specifies the data format of the major tick mark labels on the axis.



To specify a format, first select a category from the Category box, then select the required format of this category from the Format box. The selected format will be displayed in the Properties text box. You can further edit the format if you want. Then, click the **Add** button to add the format to the Stack box. Repeat the steps to specify formats for other categories. Only one format can be defined for each category.

The following are details about options in the tab:

Category

Specifies different formats from the following categories to display values on the chart.

- **Scale**
  Divides the value by hundreds, thousands, and so on.
- **Number**
  Re-formats the number value (Original example: 123456).
- **Date/Time**
  Re-formats the date/time value (Example, Wednesday, December 25, 00:00:00 GMT-08:00 2002).
- **Text**
  Specifies the length for the value.
- **Mapping**
  Maps the new value to one or more values.

Format

Lists all the formats of the selected category. For details about each format, see Data Format.

Properties

Specifies the properties for the selected format.

Sample

Displays the selected format effects.

Stack

Lists all the formats that you selected from different categories.

Add

Adds a format to the Stack box.

Remove

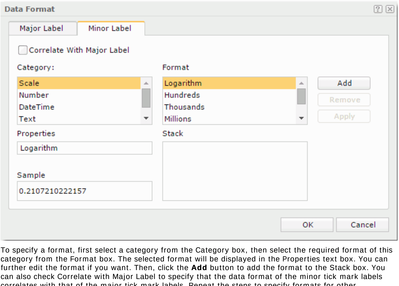
Removes a format from the Stack box.

Apply

Applies the specified format to values on the chart.

Minor Label tab

Specifies the data format of the minor tick mark labels on the axis.



To specify a format, first select a category from the Category box, then select the required format of this category from the Format box. The selected format will be displayed in the Properties text box. You can further edit the format if you want. Then, click the **Add** button to add the format to the Stack box. You can also check Correlate with Major Label to specify that the data format of the minor tick mark labels correlates with that of the major tick mark labels. Repeat the steps to specify formats for other categories. Only one format can be defined for each category.

The following are details about options in the tab:

Correlate with Major Label

If checked, the data format of the minor tick mark labels will correlate with that of the major tick mark labels automatically. Only when it is unchecked can the format properties of the minor tick mark labels take effect.

Category

Specifies different formats from the following categories to display values on the chart.

- **Scale**
  Divides the value by hundreds, thousands, and so on.
- **Number**
  Re-formats the number value (Original example: 123456).
- **Date/Time**
  Re-formats the date/time value (Example, Wednesday, December 25, 00:00:00 GMT-08:00 2002).
- **Text**
  Specifies the length for the value.
- **Mapping**
  Maps the new value to one or more values.

Format

Lists all the formats of the selected category. For details about each format, see Data Format.

Properties

Specifies the properties for the selected format.

Sample

Displays the selected format effects.

Stack

Lists all the formats that you selected from different categories.

Add

Adds a format to the Stack box.

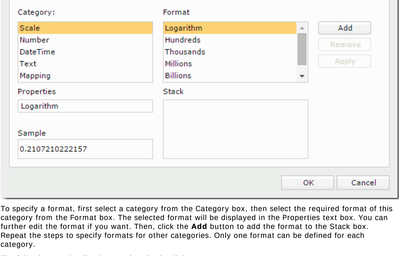
Remove

Removes a format from the Stack box.

Apply

Applies the specified format to values on the chart.

When you open the dialog to specify the data format for objects other than the category axis (the X axis) in a chart, the options in the dialog are as follows.



To specify a format, first select a category from the Category box, then select the required format of this category from the Format box. The selected format will be displayed in the Properties text box. You can further edit the format if you want. Then, click the **Add** button to add the format to the Stack box. Repeat the steps to specify formats for other categories. Only one format can be defined for each category.

The following are details about options in the dialog:

Category

Specifies different formats from the following categories to display values on the chart.

- **Scale**
  Divides the value by hundreds, thousands, and so on.
- **Number**
  Re-formats the number value (Original example: 123456).
- **Date/Time**
  Re-formats the date/time value (Example, Wednesday, December 25, 00:00:00 GMT-08:00 2002).
- **Text**
  Specifies the length for the value.
- **Mapping**
  Maps the new value to one or more values.

Format

Lists all the formats of the selected category. For details about each format, see Data Format.

Properties

Specifies the properties for the selected format.

Sample

Displays the selected format effects.

Stack

Lists all the formats that you selected from different categories.

Add

Adds a format to the Stack box.

Remove

Removes a format from the Stack box.

Apply

Applies the specified format to values on the chart.

OK

Applies the changes and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

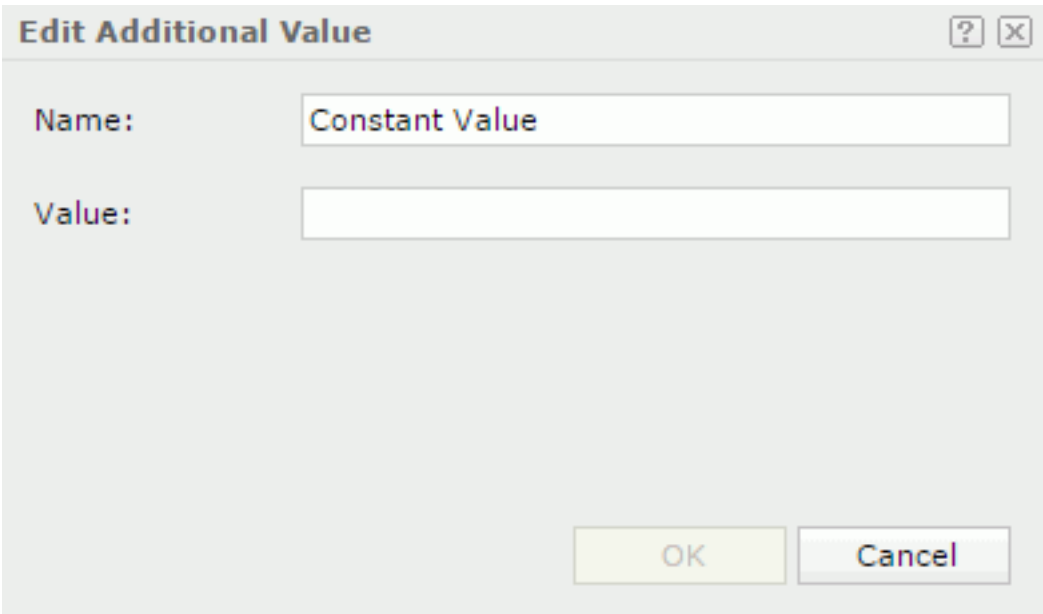
Displays the help document about this feature.

Ignores the setting and closes this dialog.

Edit Additional Value dialog

This dialog appears when you add an additional value to the Show Values box in the chart wizard. It also appears when you select an existing additional value and then click the Edit button . The dialog helps you to edit an additional value. It varies according to value types.

If it is a constant value, options in the dialog are as follows.



Name

Specifies the display name of the constant value, which will be shown under the subtype in the Show Values box.

Value

Specifies a float number for the constant value.

OK

Applies the settings and closes the dialog.

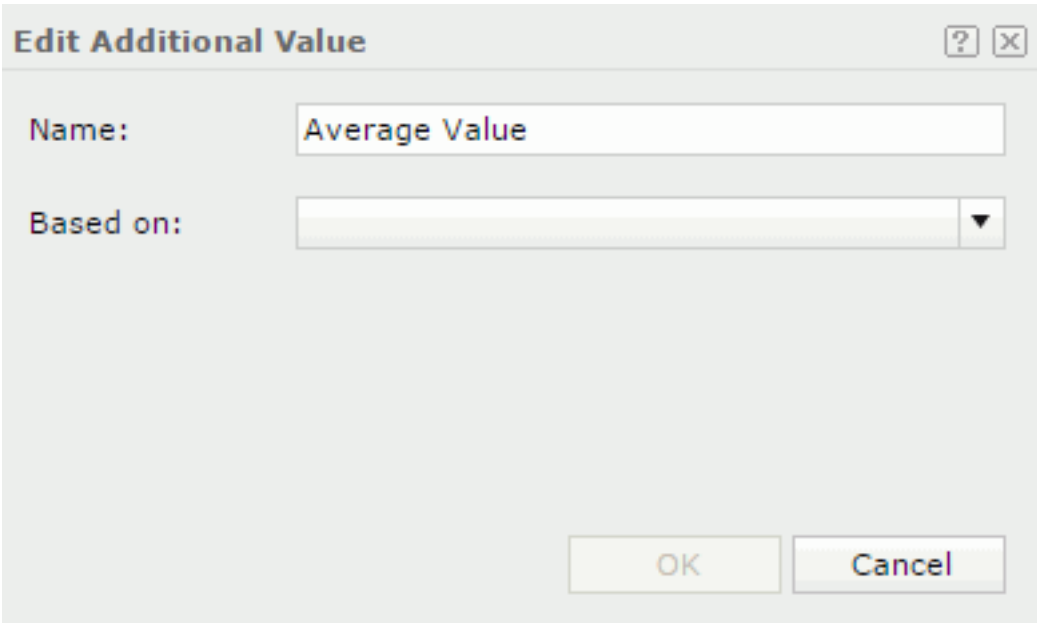
Cancel

Closes the dialog and discards any changes.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

If it is an average value, options in the dialog are as follows.



Name

Specifies the display name of the average value, which will be shown under the subtype in the Show Values box.

Based On

Specifies the aggregation object based on which the average value will be calculated.

OK

Applies the settings and closes the dialog.

Cancel

Closes the dialog and discards any changes.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Edit Conditions dialog

The dialog appears when you click the button + or in the Conditional Formatting dialog, or click the button + or in the Insert Link dialog or Edit Link dialog if the Conditional Link checkbox in the dialog is checked. It helps you to add a new condition or edit an existing condition, and has the following two modes:

- Basic
- Advanced

Advanced / Basic

Switches the dialog to the advanced/basic mode.

OK

Applies the settings and exits the dialog.

Cancel

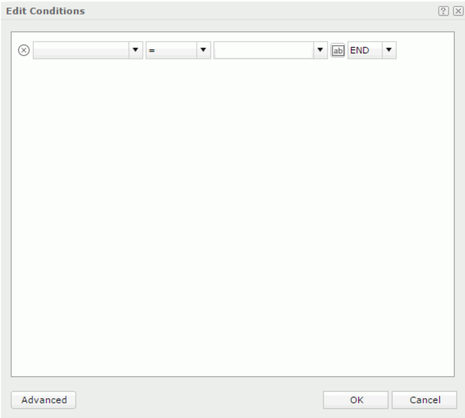
Cancels the settings and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Basic mode

The basic mode provides function for creating simple filter conditions which are connected by AND and OR operators.



Delete the current condition line.

Field

Specifies the field to be filtered.

Operator

Specifies the operator to compose the filter expression.

- **=**
  Equal to
- **>**
  Greater than
- **>=**
  Greater than or equal to
- **<**
  Less than
- **<=**
  Less than or equal to
- **!=**
  Not equal to
- **[not] in**
  Causes an enumerated list of values to appear in the WHERE clause predicate, used for evaluating for a true condition.
- **[not] like**
  Like string pattern matching operator is used to compare the first expression string value to the pattern string (the second expression). If you want to use wildcard character in the pattern string, only "_" and "%" are supported.
- **[not] between**
  Allows the system to evaluate whether or not data values are located between a range of values indicated in the predicate.
- **is [not] null**
  It is used in WHERE clause predicates to match null values occurring in a specified data field.

Value

Specifies the value of how to filter the field. You can either enter the value manually in the text box or select a value from the drop-down list. When you type in the value manually, if multiple values are required, they should be separated with ",", and if "," or "\" is contained in the values, write it as "\," or "\\".

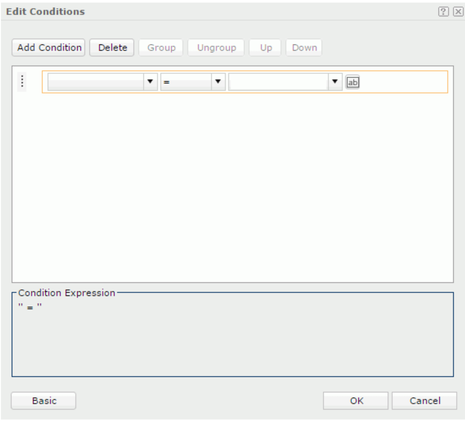
Logic

Lists the logic operator.

- **AND**
  Logic operator AND which is applied to this and the next line.
- **OR**
  Logic operator OR which is applied to this and the next line.

Advanced mode

The advanced mode enables the building of more complex filter conditions via the grouping of conditions.



Add Condition

Adds a new condition line.

Delete

Deletes the selected condition line.

Group

Makes the selected conditions in a group. Conditions can also be added to an existing group by selecting the conditions and the group while holding the Ctrl button, and then clicking the Group button.

Ungroup

Makes the selected condition ungrouped.

Up

Moves the selected condition or group up to a higher level.

Down

Moves the selected condition or group down to a lower level.

Logic

Specifies the logic operator.

- **AND**
  Logic operator AND which is applied to this and the next line.
- **OR**
  Logic operator OR which is applied to this and the next line
- **AND NOT**
  Logic operator AND NOT which is applied to this and the next line.
- **OR NOT**
  Logic operator OR NOT which is applied to this and the next line.

Field

Specifies the field to be filtered.

Operator

Specifies the operator to compose the filter expression.

Value

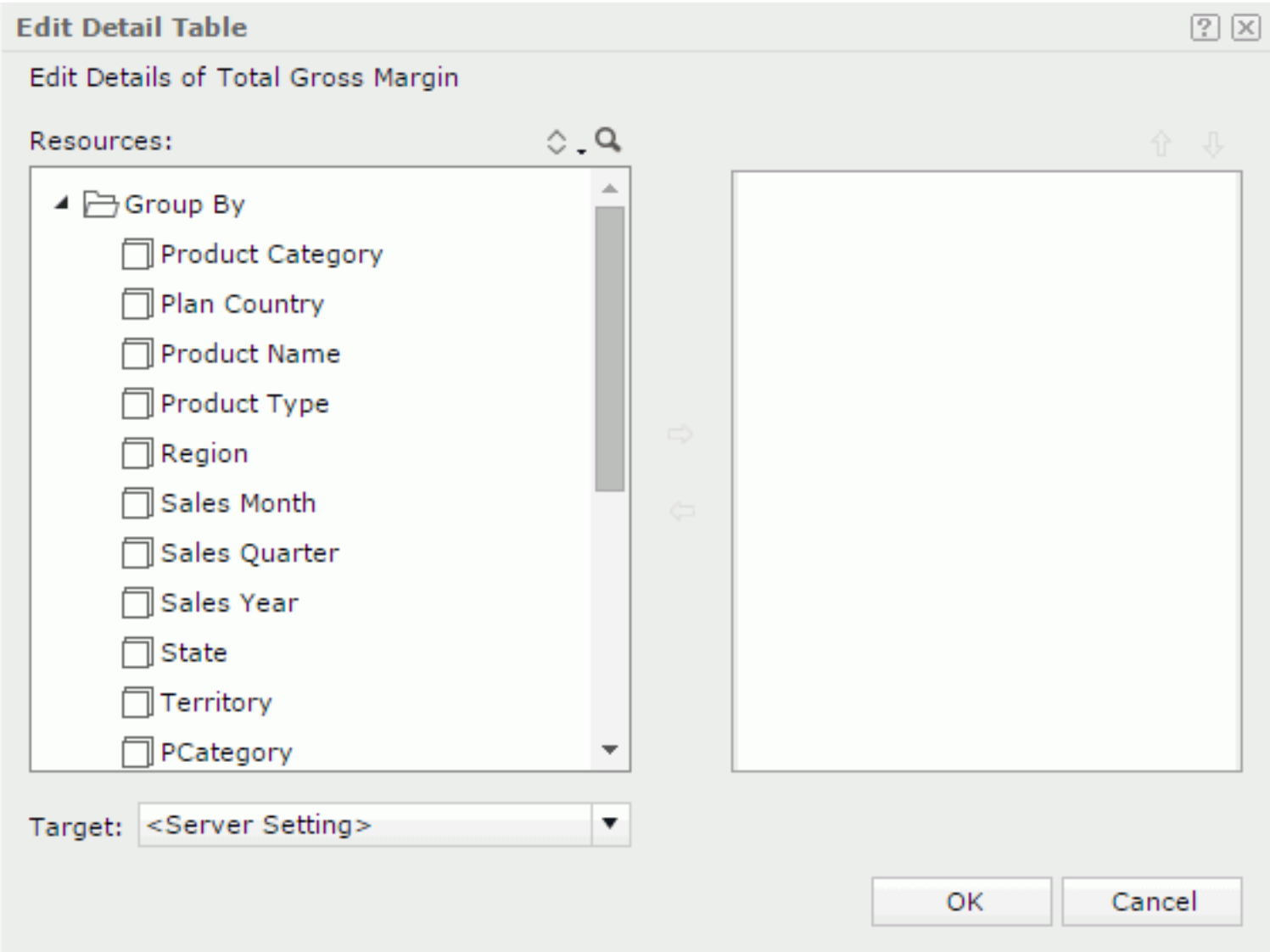
Specifies the value of how to filter the field. You can either enter the value manually in the text box or select a value from the drop-down list. When you type in the value manually, if multiple values are required, they should be separated with ",", and if "," or "\" is contained in the values, write it as "\," or "\\".

Condition Expression

Displays the SQL statement of the condition.

Edit Detail Table dialog

This dialog appears when you right-click a summary and select Edit Detail Table from the shortcut menu. It helps you to edit which fields will be displayed in the table when performing the go to detail action on the summary.



Resources

Displays all the group and detail objects in the selected business view.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.
-
  Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
-
  Highlights the previous matched text.
-
  Highlights the next matched text.

Adds the selected view element to be displayed in the table.

Removes the selected view element.

Field

Lists the view elements that have been added to the table.

Moves the selected view element one step up.

Moves the selected view element one step down.

Target

Specifies the window or frame in which to load the detail table.

- **New Window**
  Loads the detail table into a new window.
- **Same Frame**
  Loads the detail table into the same frame or window where the main report is.
- **Other Frame**
  Loads the detail table into some other available frame. Type the frame name into the text box to find the frame. If the frame does not exist, the detail table report will be loaded into a new window.
- **<Server Setting>**
  Loads the detail table according to setting of the Target of Detail Table Report and Links option in the Customize Profile panel of the Profile dialog.

OK

Applies the changes and closes the dialog.

Cancel

Cancels the changes and exits the dialog.

Displays the help document about this feature.

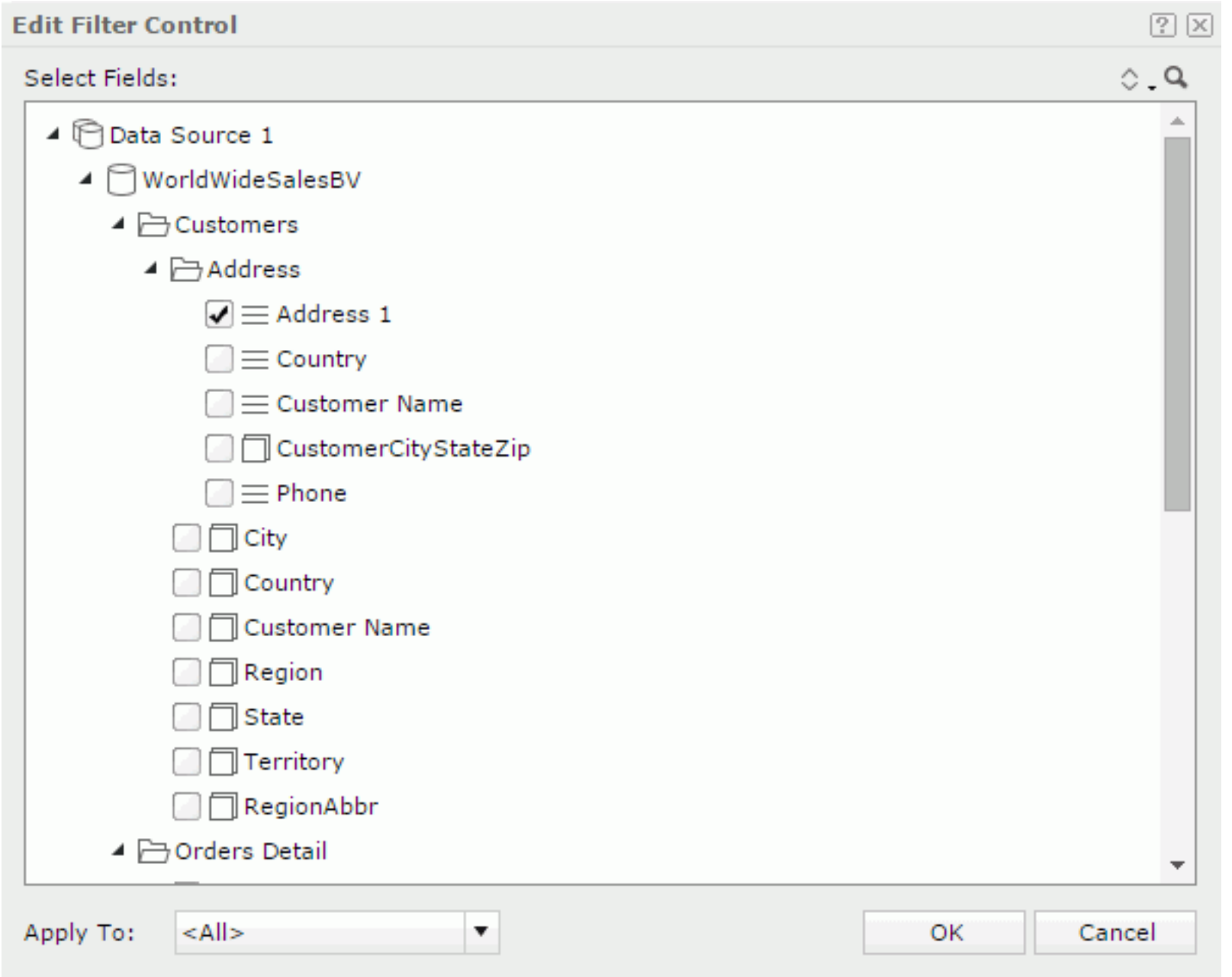
Ignores the setting and closes this dialog.

Related topics:

- Going through the report data

Edit Filter Control dialog

This dialog appears when you right-click on a filter control and then select Edit. It helps you to edit the filter control.



Select Fields

Specifies the fields to bind to the filter control. All the selected fields should be of the same data type. All uncomparable data type fields cannot be bound to filter controls, such as Binary, Blob, Clob, Longvarchar, Longvarbinary, and Varbinary.

The common usage is to select one field, and then based on the field to filter the data of the components created from the same data source as the field.

To filter components using different data sources, choose a common field all the data sources contain and select the field in all the data sources.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.
-
  Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
-
  Highlights the previous matched text.
-
  Highlights the next matched text.

Apply To

Specifies the components to which the filter created with the filter control will be applied.

OK

Inserts a filter control into the report and closes the dialog.

Cancel

Cancels the insertion and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Edit Image dialog

This dialog appears when you right-click an image and select Edit from the shortcut menu. It helps you to edit the image.

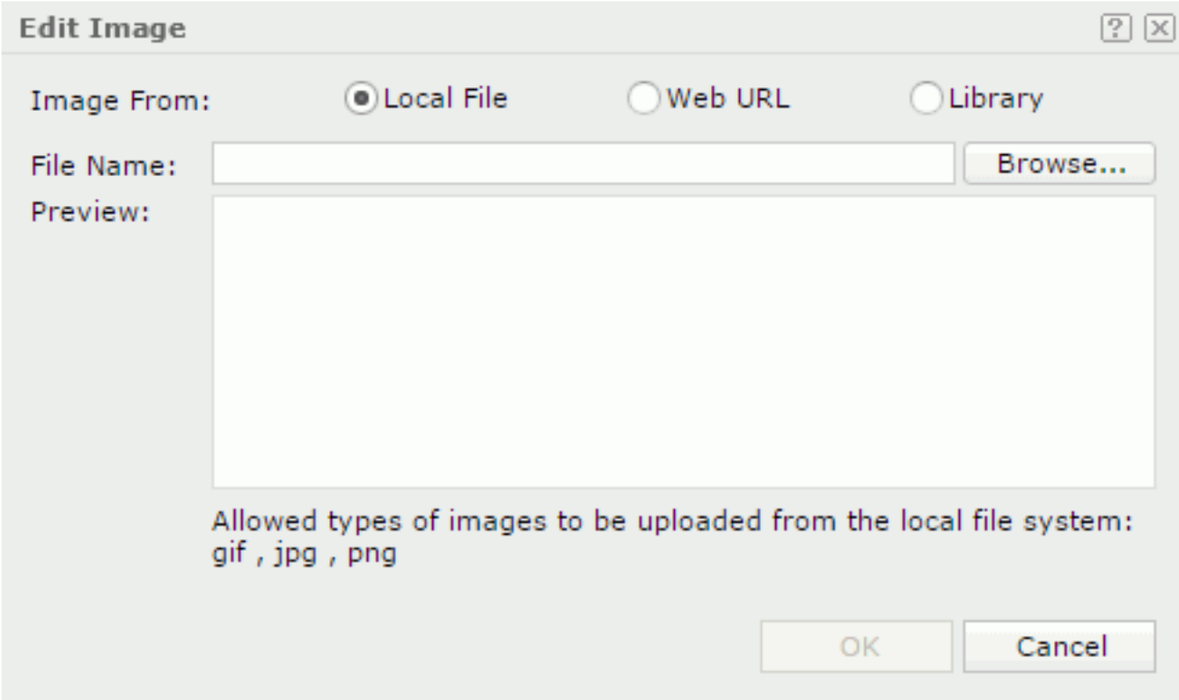


Image From

Specifies the source of the image file.

- **Local File**
  Specifies to use an image from the local file system.
  - **File Name**
    Specifies the path and name of the image file. You can click Browse button to locate the image file.
- **Web URL**
  Specifies to use an image via URL.
  - **Image URL**
    Specifies the URL of the image file.

    Note: If your JReport Server is in an intranet which requires a proxy, to successfully access the image via URL, you need to add the parameters `-Dhttp.proxyHost=XXX -Dhttp.proxyPort=XX` to the server's startup file JRServer.bat, which is located in `<install_root>\bin`.
- **Library**
  Specifies to use an existing image.
  - **My Images**
    The My Images folder is a virtual location where JReport Server stores the images that have once been inserted into reports. Select the one you want to use.

Preview

Displays a preview of the selected image.

OK

Applies the settings and closes the dialog.

Cancel

Cancels the changes and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Edit Link dialog

This dialog appears when you select an object which contains the defined link, right-click it and select Edit Link on the shortcut menu. In helps you to edit the link on the object.

Conditional Link

Specifies whether it is a conditional link. If checked, the object can be linked to different targets based on different conditions.

Adds a new condition in the Edit Conditions dialog.

Edits the selected condition.

Removes the selected condition.

Moves the selected condition one step up.

Moves the selected condition one step down.

- **Others**
  If checked, you can define to which target the object will be linked when none of the conditions you have specified is met.

Link Type

Specifies the type of the link target for the object if it is not a conditional link or for the object under the selected condition if it is a conditional link. It can be one of the following

- Report
- URL
- E-mail
- Content

More

Displays more link options. Available only for the Report and Content link types.

OK

Applies the settings and closes the dialog.

Cancel

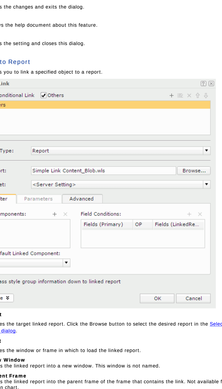
Cancels the changes and exits the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Link to Report

It helps you to link a specified object to a report.



Report

Specifies the target linked report. Click the Browse button to select the desired report in the Select a Report dialog.

Target

Specifies the window or frame in which to load the linked report.

- **New Window**
  Loads the linked report into a new window. This window is not named.
- **Parent Frame**
  Loads the linked report into the parent frame of the frame that contains the link. Not available for link in chart.
- **Same Frame**
  Loads the linked report into the same frame as the link.
- **Whole Window**
  Loads the linked report into the full browser window. Not available for link in chart.
- **<Server Setting>**
  Loads the linked report according to setting of the Target of Detail Table Report and Links option in the Customize Profile panel of the Profile dialog.
- **Other Frame**
  Loads the linked report into some other specified frame. If the frame name does not exist, the linked file will be loaded into a new window.

Filter tab

Specifies the filter conditions based on which the link relationship between the primary report and the linked report is set up.

- **Components**
  Specifies the components in the linked report that will be interlinked with the primary report.
  - Adds a component in the linked report to be interlinked with the primary report.
  - Removes the selected component.
- **Default Linked Component**
  Specifies the default linked component in the linked report.
- **Field Conditions**
  Lists the link conditions between the primary report and the linked report.
  - Adds a new condition line.
  - Removes the selected condition line.
  - **Fields(Primary)**
    Lists the selected fields of the primary report.
  - **OP**
    Specifies the operator to compose the condition. It can be "=", "<>", "<", ">", "<=", ">=" or "IN".
  - **Fields(LinkedReport)**
    Lists the fields in the datasets of the linked report which are of the same data type as the selected fields in the primary report.

Parameters tab

This tab is available when the linked report uses parameters. It helps you to assign values to the parameters of the linked report automatically.

- **Parameters**
  Lists all parameters contained in the linked report.
- **Value**
  Lists all the DBFields, formulas, summaries and parameters in the primary report which are of the same data type as the parameters of the linked report. Select the field the value of which you want to assign to parameter of the linked report from the drop-down list.

Advanced tab

Specifies to pass the values of the filter objects such as filter controls and sliders in the primary report to those in the linked report.

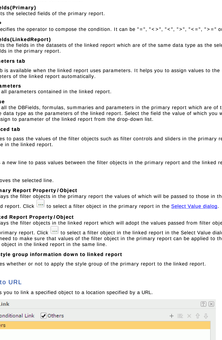
- Adds a new line to pass values between the filter objects in the primary report and the linked report.
- Removes the selected line.
- **Primary Report Property/Object**
  Displays the filter objects in the primary report the values of which will be passed to those in the linked report. Click to select a filter object in the primary report in the Select Value dialog.
- **Linked Report Property/Object**
  Displays the filter objects in the linked report which will adopt the values passed from filter objects in the primary report. Click to select a filter object in the linked report in the Select Value dialog. You need to make sure that values of the filter object in the primary report can be applied to the filter object in the linked report in the same line.

Pass style group information down to linked report

Specifies whether or not to apply the style group of the primary report to the linked report.

Link to URL

It helps you to link a specified object to a location specified by a URL.



Hyperlink

Specifies the URL for the hyperlink that is to be used to link the object.

- **Add Dynamic Field**
  Opens the Select Field dialog to insert a field into the URL.

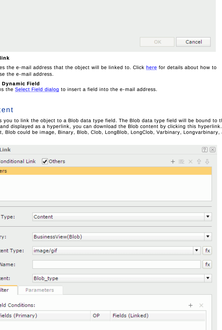
Target

Specifies the window or frame in which to load the location specified by the URL.

- **New Window**
  Loads the linked file into a new window. This window is not named.
- **Parent Frame**
  Loads the linked file into the parent frame of the frame that contains the link.
- **Same Frame**
  Loads the linked file into the same frame as the link.
- **Whole Window**
  Loads the linked file into the full browser window.
- **<Server Setting>**
  Loads the linked file according to setting of the Target of Detail Table Report and Links option in the Customize Profile panel of the Profile dialog.
- **Other Frame**
  Loads the linked file into some other specified frame. If the frame name does not exist, the linked file will be loaded into a new window.

Link to E-mail

It helps you to link the specified object to an e-mail address.



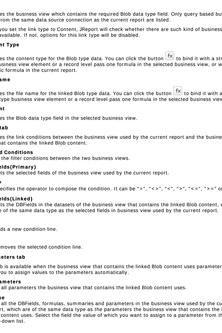
Hyperlink

Specifies the e-mail address that the object will be linked to. Click here for details about how to compose the e-mail address.

- **Add Dynamic Field**
  Opens the Select Field dialog to insert a field into the e-mail address.

Content

It helps you to link the object to a Blob data type field. The Blob data type field will be bound to the object and displayed as a hyperlink; you can download the Blob content by clicking this hyperlink. In Jreport, Blob could be image, Binary, Blob, Clob, LongBlob, LongClob, Varbinary, Longvarbinary, and so on.



Query

Specifies the business view which contains the required Blob data type field. Only query based business views from the same data source connection as the current report are listed.

When you set the link type to Content, Jreport will check whether there are such kind of business views available. If not, options for this link type will be disabled.

Content Type

Specifies the content type for the Blob type data. You can click the button to bind it with a string type business view element or a record level pass one formula in the selected business view, or with a dynamic formula in the current report.

File Name

Specifies the file name for the linked Blob type data. You can click the button to bind it with a string type business view element or a record level pass one formula in the selected business view.

Content

Specifies the Blob data type field in the selected business view.

Filter tab

Specifies the link conditions between the business view used by the current report and the business view that contains the linked Blob content.

- **Field Conditions**
  Lists the filter conditions between the two business views.
  - **Fields(Primary)**
    Lists the selected fields of the business view used by the current report.
  - **OP**
    Specifies the operator to compose the condition. It can be "=", "<>", "<", ">", "<=", ">=" or "IN".
  - **Fields(Linked)**
    Lists the DBFields in the datasets of the business view that contains the linked Blob content, which are of the same data type as the selected fields in business view used by the current report.
  - Adds a new condition line.
  - Removes the selected condition line.

Parameters tab

This tab is available when the business view that contains the linked Blob content uses parameters. It helps you to assign values to the parameters automatically.

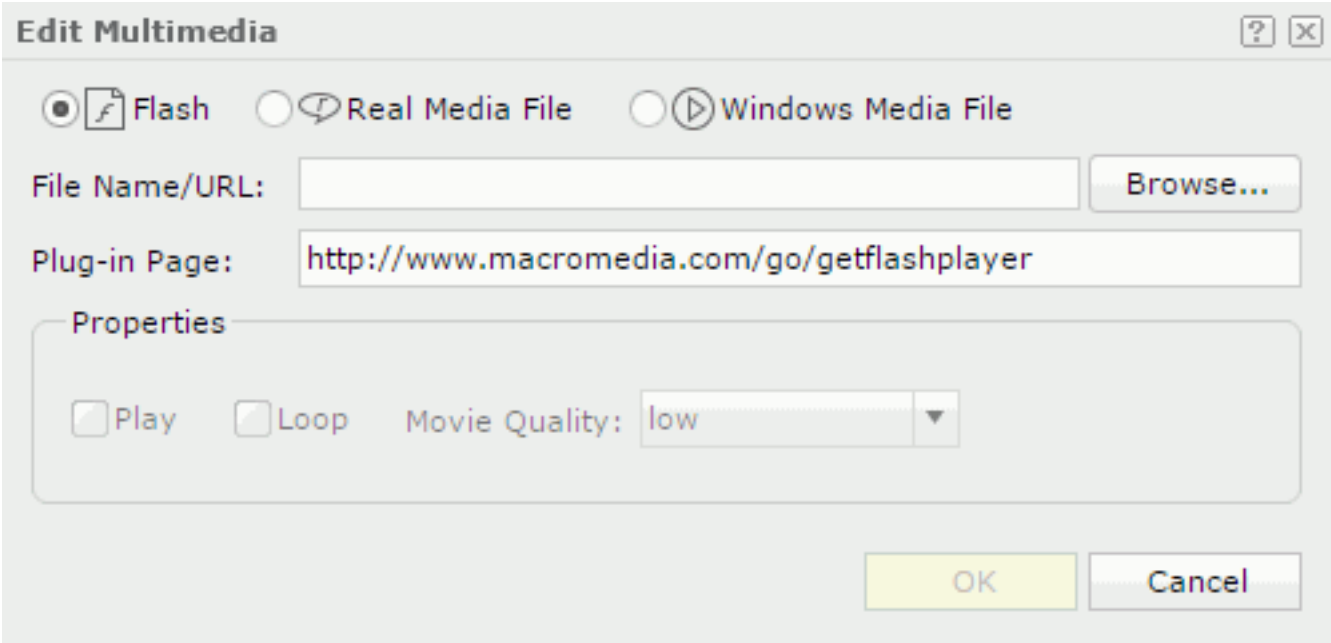
- **Parameters**
  Lists all parameters the business view that contains the linked Blob content uses.
- **Value**
  Lists all the DBFields, formulas, summaries and parameters in the business view used by the current report, which are of the same data type as the parameters the business view that contains the linked Blob content uses. Select the field the value of which you want to assign to a parameter from the drop-down list.

Related topics:

- Binding links to objects

Edit Multimedia dialog

This dialog appears when you right-click a multimedia object and select Edit from the shortcut menu. It helps you to edit the multimedia object.



Flash

Specifies to insert a flash file to the current report.

Real Media File

Specifies to insert a realmedia file to the current report.

Windows Media File

Specifies to insert a Windows Media file to the current report.

File Name / URL

Specifies the name or URL of the multimedia object that is to be inserted into the report. Type in the name or URL in the text box or click Browse to find the file.

Plug-in Page

Specifies the URL of the plug-in page from which to download the player with which to play the multimedia object file (if it isn't already installed on your local disk).

Properties

Specifies the properties of the multimedia object.

- **For Flash**
  - **Play**
    Specifies whether to start the flash file automatically when the report is opened.
  - **Loop**
    Specifies whether to play the flash file repeatedly.
  - **Movie Quality**
    Specifies the quality of the flash.
- **For Real Media File**
  - **Auto Start**
    Specifies whether to start the realmedia file automatically when the report is opened.
  - **Center**
    Specifies whether to place the realmedia file in the center of the report.
  - **Loop**
    Specifies whether to play the realmedia file repeatedly.
  - **Loop Number**
    Specifies how many times you want to play the realmedia file.
- **For Windows Media File**
  - **Auto Start**
    Specifies whether or not to start the Windows Media automatically when the report is opened.
  - **Auto Rewind**
    Specifies whether or not to rewind the Windows Media file automatically.
  - **Play Count**
    Specifies the number of times you want to play this media file.

OK

Applies the settings for the multimedia object and closes this dialog.

Cancel

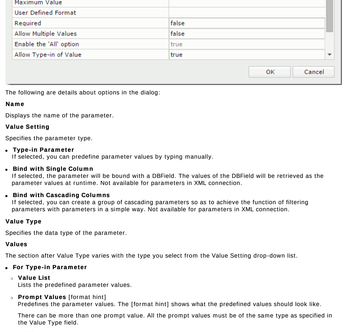
Cancels the edition and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Edit Parameter dialog

The dialog appears when you click the button in the Value text field of the Query Filter dialog or Filter dialog, select a local parameter from the Local Parameters node of the drop-down list, right-click it and select Edit from the contextual menu. It helps you to edit a local parameter for the report.



The following are details about options in the dialog:

Name

Displays the name of the parameter.

Value Setting

Specifies the parameter type.

- **Type-in Parameter**
  If selected, you can predefine parameter values by typing manually.
- **Bind with Single Column**
  If selected, the parameter will be bound with a DBField. The values of the DBField will be retrieved as the parameter values at runtime. Not available for parameters in XML connection.
- **Bind with Cascading Columns**
  If selected, you can create a group of cascading parameters so as to achieve the function of filtering parameters with parameters in a simple way. Not available for parameters in XML connection.

Value Type

Specifies the data type of the parameter.

Values

The section after Value Type varies with the type you select from the Value Setting drop-down list.

- **For Type-in Parameter**
  - **Value List**
    Lists the predefined parameter values.
  - **Prompt Values** [format hint]
    Predefines the parameter values. The [format hint] shows what the predefined values should look like.
    There can be more than one prompt value. All the prompt values must be of the same type as specified in the Value Type field.
    - +
      Adds a new prompt value to the list. Double-click in the value line to edit the value. If the parameter is of Date, DateTime, or Time type, you can also click the calendar button to set a date and time value in the Calendar dialog.
    - ×
      Removes the selected prompt value.
    - ⇧
      Adjusts the display sequence of selected value by moving it up a step.
    - ⇩
      Adjusts the display sequence of selected value by moving it down a step.
- **For Bind with Single Column**
  JReport provides such a mechanism that links each field value of Display Column with the exact value of Bind Column. When you select the field value from Display Column, the field value of Bind Column is actually sent to the query and filters the query result. It helps you to bind the parameter to a DBField (column) in order to provide a list of DBfield values for users to select at runtime. The selected value of the bound DBField will be used as the parameter value. Additionally, instead of displaying the values of the bound one, you can also choose to display values of other DBfield, which probably makes more sense to the report end users. For example, it might be confusing if you provide a list of customer ID numbers for users to select at runtime, since the ID numbers could probably mean nothing to them. In cases like this, it is wise for you to display values of other fields which could probably make more sense. For this case, you might prefer to display customer names instead of ID numbers. When the report end users select a customer name from the list, its ID number is passed to the query as the parameter value so that the search criteria can be fulfilled. So, before the report with a parameter bound to a DBField is processed, a query statement will be executed first to retrieve the values of the bound field and display field.
  - **Source**
    Specifies the business view in the current data source from which to retrieve the columns which can be used to bind with the parameter. Select one from the drop-down list.
  - **Bind Column**
    Specifies the DBfield to be bound with the parameter.
  - **Display Column**
    When viewing a report which contains the parameter, the field values related to the column name you selected in the drop-down list of Display Column will be displayed in the Enter Parameter Values dialog at runtime.
- **For Bind with Cascading Columns**
  - **Source**
    Specifies the business view in the current data source from which to retrieve the columns which can be used to bind with the parameters. Select one from the drop-down list.
  - **Value List**
    Defines a group of cascading parameters.
    - +
      Adds a new parameter.
    - ×
      Removes the selected parameter.
    - ⇧
      Adjusts the parameter level by moving it up a step. The higher the position is, the higher the level is. The lower level field values are controlled by the higher level field values.
    - ⇩
      Adjusts the parameter level by moving it down a step.
  - **Bind Column**
    Specifies the DBfield to be bound with the parameter.
  - **Display Column**
    When viewing a report which contains the parameter, the field values related to the column name you selected in the drop-down list of Display Column will be displayed in the Enter Parameter Values dialog at runtime.
  - **Parameter**
    Specifies whether to set a group of Bind Column and Display Column as a parameter which will become a member of the cascading parameter group. All parameters in the cascading group will be created as independent parameters in the catalog after you finish the dialog.
    The parameter name is provided by JReport automatically, and the format is *[The name specified as parameter name]-[Bind Column name]*.

Options

Specifies options for the parameter. The options vary with the type you select from the Value Setting drop-down list.

- **Prompt Text**
  Specifies the text for prompting users to provide the value of the parameter.
  Tips:
  - To help users avoid typing a value out of range, you can add the value range to the end of the prompt text. For example, "Type a customer ID (1~100):".
  - To help users avoid typing a value in a wrong pattern, you can add the value pattern to the end of the prompt text. For example, "Type an order date (MMM-dd-yy):".
- **Minimum Value/Length**
  Specifies the minimum value allowed for the parameter. If the parameter value type is String, this option is defined as Minimum Length, which means that you can set the minimum allowed length of the string.
- **Maximum Value/Length**
  Specifies the maximum value allowed for the parameter. If the parameter value type is String, this option is defined as Maximum Length, which means that you can set the maximum allowed length of the string.
- **String Encoding**
  Specifies the encoding for the parameter. Available only when the parameter's data type is defined as String.
- **User Defined Format**
  For certain types of parameters, the value format supplied by JReport may not satisfy your requirement. In this case, you can define your own preferred format. When you have defined the value format, all the values you supply at runtime should be based on it. Otherwise there will be an error message.
  Notes:
  - The letters that stand for the user defined format should follow the JDK standard. In addition, the user defined format only limits the format of the input value, which does not mean that the display value will be in the same format. To change the display format, you will need to specify the Format property of this parameter in the Report Inspector.
  - The user defined format will show different appearance under different locale, and you need input values according to the displayed format. For example, under the Spanish or French locale, ###.## will be displayed as #0,## and you should follow #0,## to input values.
- **Required**
  If set to true, the parameter will be treated as a required parameter. A required parameter must be fulfilled, which means that you must supply a value in the Enter Parameter Values dialog before the report can be run. If set to false, the parameter will be treated as an optional parameter. An optional parameter can be left blank, which means you can either type a value, or leave it as it is.
  In the Enter Parameters Value dialog, the prompt text of a required parameter will be marked with a * sign.
  Note: If the parameter is optional and you do not expect it to take effect in your report, in the Enter Parameter Values dialog, clear the content in the value field. Otherwise, any value left in the field will function in reports.
- **Hide Parameter When Single Value Returned**
  Specifies to hide the parameter at runtime when only a single value is returned for the parameter. The returned single value will be used as default value of the parameter.
- **Allow Multiple Values**
  Enables multiple parameter values to be selected. If it is set to true, when running a report with the parameter, the Enter Values dialog is provided for you to specify multiple values.
  Note: Multi-value parameters can only be supported in dataset filters, query filters, imported SQL files, and subreport/link report conditions.
- **Enable the "All" Option**
  Displays the All checkbox in the Enter Values dialog which means that the parameter value result is all the values that match the parameter data type. Available when Allow Multiple Values is set to true.
- **Treat Blank as Null**
  Available only when the parameter's data type is defined as String. If it is true, the parameter value will be null; if false, the default parameter value is "".
- **Use Current Date-Time**
  If set to true, the current date time in your system will be used as value of the parameter. Available only when the parameter's data type is defined as Date, Time or DateTime.
- **Use Current Date-Time When Blank**
  If set to true, when the value of the parameter is blank, the current date time in your system will be used as value of the parameter. Available only when the parameter's data type is defined as Date, Time or DateTime.
  If the type of the parameter is set to Date, Time or DateTime, you can choose whether to get parameter value dynamically from the system's date time by enabling one of the above options - Use Current or Use Current When Blank. However, if you want to use this feature, you need to make sure your system's DateTime format is set to yyyy-MM-dd H:mm:ss as it is the only format JReport supports; otherwise, you need to make use of the User Defined Parameter Format feature. Also, if one of the two options is enabled, you need to provide the parameter's prompt value according to the data type of your parameter.
  - When the data type of the parameter is Date, you can specify the value according to the format yyyy-MM-dd.
    - yyyy-01-01: Year will be dynamically changed.
    - 2007-MM-01: Month will be dynamically changed.
    - 2007-01-dd: Date will be dynamically changed.
    - yyyy-MM-01: Year and month will be dynamically changed.
    - yyyy-01-dd: Year and date will be dynamically changed.
    - yyyy-MM-dd: Year, month, and date will be dynamically changed.
    - 2007-MM-dd: Month and date will be dynamically changed.
  - When the data type of the parameter is Time, you can specify the value according to the format H:mm:ss.
    - H:00:00: Hour will be dynamically changed.
    - 12:mm:00: Minute will be dynamically changed.
    - 12:00:ss: Second will be dynamically changed.
    - H:mm:00: Hour and minute will be dynamically changed.
    - 12:mm:ss: Minute and second will be dynamically changed.
    - H:00:ss: Hour and second will be dynamically changed.
    - H:mm:ss: Hour, minute and second will be dynamically changed.
  - When the data type of the parameter is DateTime, specify the value in a combination of the above.
- **Sort**
  Specifies how to sort values of cascading parameters.
  - **Ascend**
    Sorts parameter values in an ascending order.
  - **Descend**
    Sorts parameter values in a descending order.
  - **No Sort**
    The parameter values are displayed in the original order as in database.
- **Sort By**
  Sorts parameter values by sorting other fields in the same data source as the cascading group. When Sort By is selected, the Custom Sort dialog will be displayed for specifying fields and their sorting orders.
- **Distinct**
  If set to true, when viewing a report that contains the parameter, in the Enter Parameter Values dialog, identical values will appear only once and no duplicate values will be listed.
- **Allow Type-in of Value**
  Specifies whether to enable typing values besides selecting values when setting parameter values. The property is enabled for Type-in Parameter by default.
- **When Out of Value Range**
  Specifies how JReport will respond if the typed value is out of the predefined value list. The property is enabled only when the option Allow Type-in of Value is set to true.
  There are three solutions as follows:
  - **Adopt directly**
    The typed value will be used directly.
  - **Use default**
    The default value defined when creating the parameter will be used.
  - **Warn**
    Displays a warning message telling that "Value xxx is out of the available value list".
    Note: For a multi-value parameter, if one of the input values is not in the value list, the whole input result will be treated as out of range.
- **Get Value from API Only**
  Specifies whether the parameter values can be input via GUI or not.
  If true, the parameter values should only be obtained from API and there will be no parameter dialogs available on any JReport Designer or Server GUI for inputting values. Users will have to give the values of the parameter by API such as URLs, coding, and sessions, otherwise the default value will be used.
  If false, the parameter can be obtained from both GUI and API.
  If a member of a cascading parameter group is set to get values from API only, it means that all the higher level members shall get values from API only. This also applies to parameters that some other parameters depend on. If in JDashboard several parameters share values, when any of them is set to get values from API only, they will share the values from API.
- **On Parameter Value Change**
  Specifies to define an action that will be called when the parameter value changes. Choose a formula from the drop-down list to develop an action, which will be called before the other actions defined in the report or library component. It can work as a parameter value validation rule.

OK

Applies all changes and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Displays the help document about this feature.

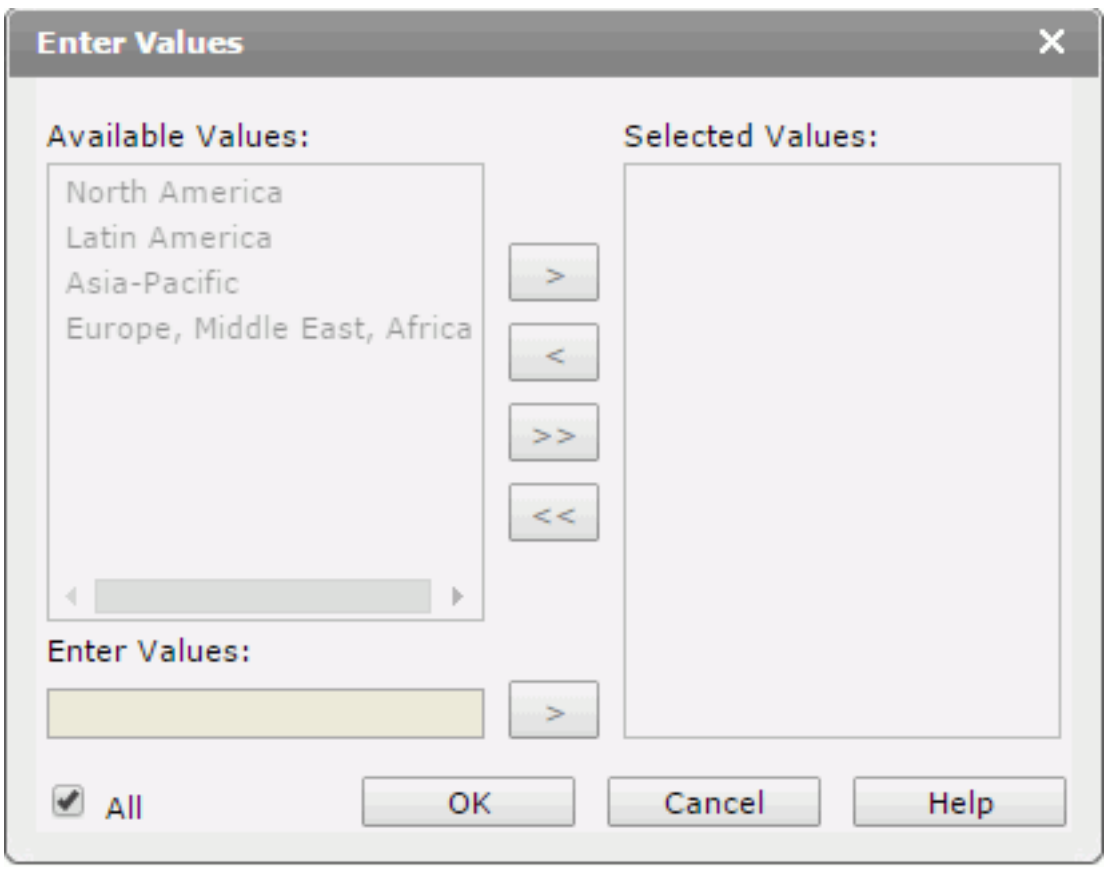
Ignores the setting and closes this dialog.

Related topics:

- Applying parameters

Enter Values dialog

The dialog appears when you click the button ... while specifying values for a parameter. It helps you to specify multiple values for the parameter.



Available Values

Lists all predefined parameter values for selection. When the parameter is bound with a column, but the Display Column is different from the Bind Column, values of the Display Column are listed here.

Selected Values

Lists the values selected. The selected values are case sensitive.

Adds the selected values from the Available Values box to the Selected Values box.

Removes the selected values from the Selected Values box.

Adds all the selected values from the Available Values box to the Selected Values box.

Removes all the selected values from the Selected Values box.

Enter Values

This option is available when the parameter's Allow Type-in of Value property is set to true in JReport Designer.

Enter a value manually in the text box and then click the button next to add the value to the Selected Values box. When the parameter is bound with a column, but the display column is different from the bind column, make sure the value you enter is that of the bind column.

Opens the Calendar dialog to specify a date and time value. Available only for Date, DateTime, or Time parameter which allows for type-in values and is not bound with any column.

Search

This option is available when the parameter's Allow Type-in of Value property is set to false in JReport Designer. It is used to search for values among the available values. After you type text in the Search text box, the matched text in the available values will be highlighted.

All

If selected, it means that the parameter value result is all the values in the DBMS. This option is available when the parameter's Enable the "All" Option property is set to true in JReport Designer. This is translated in SQL to remove the parameter which may select more values than listed in the available values list.

For example, when the parameter query is:

```
SELECT CUSTOMERS.CUSTOMERID,CUSTOMERS.CUSTOMERNAME FROM CUSTOMERS where CUSTOMERS.
CUSTOMERID>0 and CUSTOMERS.CUSTOMERID<4
```

When you click >> to add 1, 2 and 3 as the parameter values, which are all the available values, the SQL is:

```
(CUSTOMERS.CUSTOMERID IN ( 1,2,3))
```

Run the same report again and this time check the All checkbox in the Enter Values dialog, the query is then:

```
( 1 = 1)
```

In this case, you will get more customers even though available values are only 1 •3.

Note: When a multi-value parameter is inserted as a field into a report and All is selected as the value, the field will show the string "All". If the parameter allows type-in values, "All" means all possible values of the parameter data type; if the parameter does not, "All" represents all the values of the parameter that come from the DBMS.

OK

Applies the selected values to the parameter and closes the dialog.

Cancel

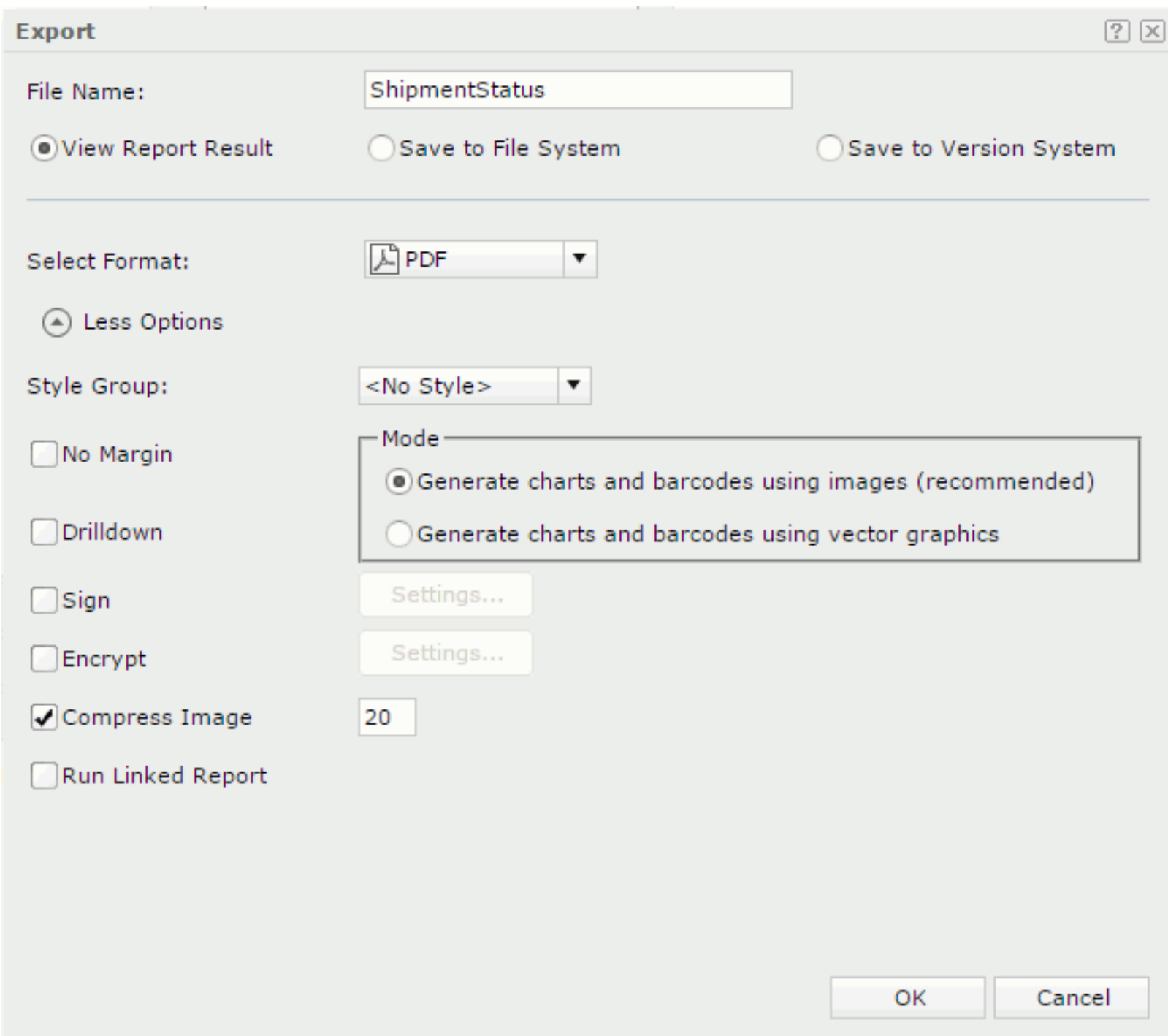
Does not retain any changes and closes the dialog.

Help

Displays the help document about this feature.

Export dialog

This dialog appears when you click Menu > File > Export. It helps you to set settings for exporting the report result to different formats.



File Name
Specify a name for the result file in the field.

View Report Result

If selected, the result will be directly opened in the web browser if the format is supported by a plug-in of the web browser; otherwise you will be prompted to save the result file.

Save to File System

If selected, the report result will be saved to a specified folder.

Save to Version System

If selected, the report result will be saved as a result version in JReport Server's versioning system.

Select Format

Specifies the format to which the report result will be exported.

More / Less Options

Click to show/hide the additional settings for exporting the report to the specified format.

- **Style Group**
  Specifies the style group that will be applied to the exported result. When the <No Style> item in the Style Group drop-down list is selected, the style group property defined for specific exporting format in JReport Designer will be applied to export the report result to that format.

- **Properties**
  Specifies the properties for the selected format:
  - PDF
  - HTML
  - Excel
  - Text
  - RTF
  - XML
  - PostScript

OK

Exports the report with the settings you specified.

Cancel

Cancels the operation and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Field Properties dialog

The dialog appears when you right-click a field and select Properties from the shortcut menu. It helps you to specify the properties of the field and contains the following tabs:

- General tab
- Font tab
- Border tab

OK

Applies the settings and closes this dialog.

Cancel

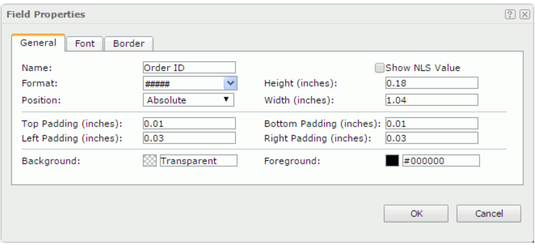
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the field.



Name

Specifies the display name of the field.

Show NLS Value

Specifies to show the translated name of the display name of the field in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the field is not modified.

Width

Specifies the width of the field.

Format

Specifies the format of the field.

Height

Specifies the height of the field.

Position

Displays the position mode of the field. If the field is directly contained in the report body, a tabular cell, or a text box, its position mode can be modified.

- **Absolute**: The field's position will be decided by its X and Y property values.
- **Static**: The field will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled.

Top Padding

Specifies the space between the text of the field and its top border.

Bottom Padding

Specifies the space between the text of the field and its bottom border.

Left Padding

Specifies the space between the text of the field and its left border.

Right Padding

Specifies the space between the text of the field and its right border.

Background

Specifies the background color of the field.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

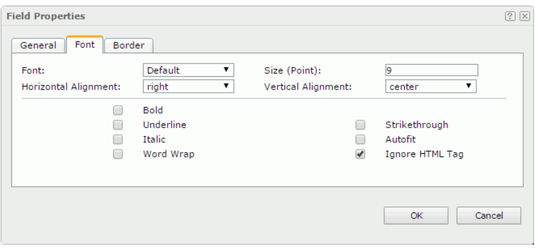
Foreground

Specifies the foreground color of the field.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB.

Font tab

This tab shows the font-related information of the field.



Font

Specifies the font face of the text.

Size

Specifies the font size of the text.

Horizontal Alignment

Specifies the horizontal alignment mode of the text in the field.

Vertical Alignment

Specifies the vertical alignment mode of the text in the field.

Bold

Specifies whether to make the text bold or not.

Underline

Specifies whether the text will be underlined or not.

Strikethrough

Specifies whether or not to attach a strikeout line to the text.

Italic

Specifies whether to make the text italic or not.

Autofit

Specifies whether or not to automatically adjust the width of the field.

Word Wrap

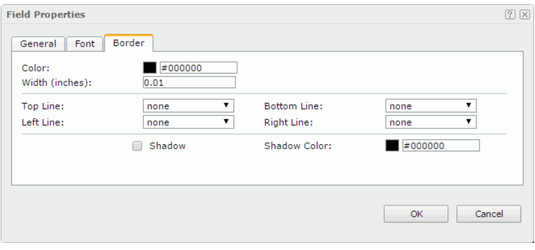
Specifies whether or not to wrap the text to the field width.

Ignore HTML Tag

If this option is unchecked, JReport will parse HTML tag elements in the field value while the report is to be saved as an HTML file; or the field value will appear in the HTML file the same as that in Web Report Studio (HTML tag elements in the field value, if any, will not be parsed).

Border tab

This tab shows information about borders of the field.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Shadow

Specifies whether the borders will have a shadow effect or not. This property cannot be rendered in Web Report Studio or JDashboard.

Shadow Color

Specifies the color of the border shadow. This property cannot be rendered in Web Report Studio or JDashboard.

Fill Effects dialog

This dialog helps you to specify either a gradient or an image as the fill effect of an object. It contains the following tabs:

- Gradient tab
- Image tab

OK

Applies the changes and closes the dialog.

Cancel

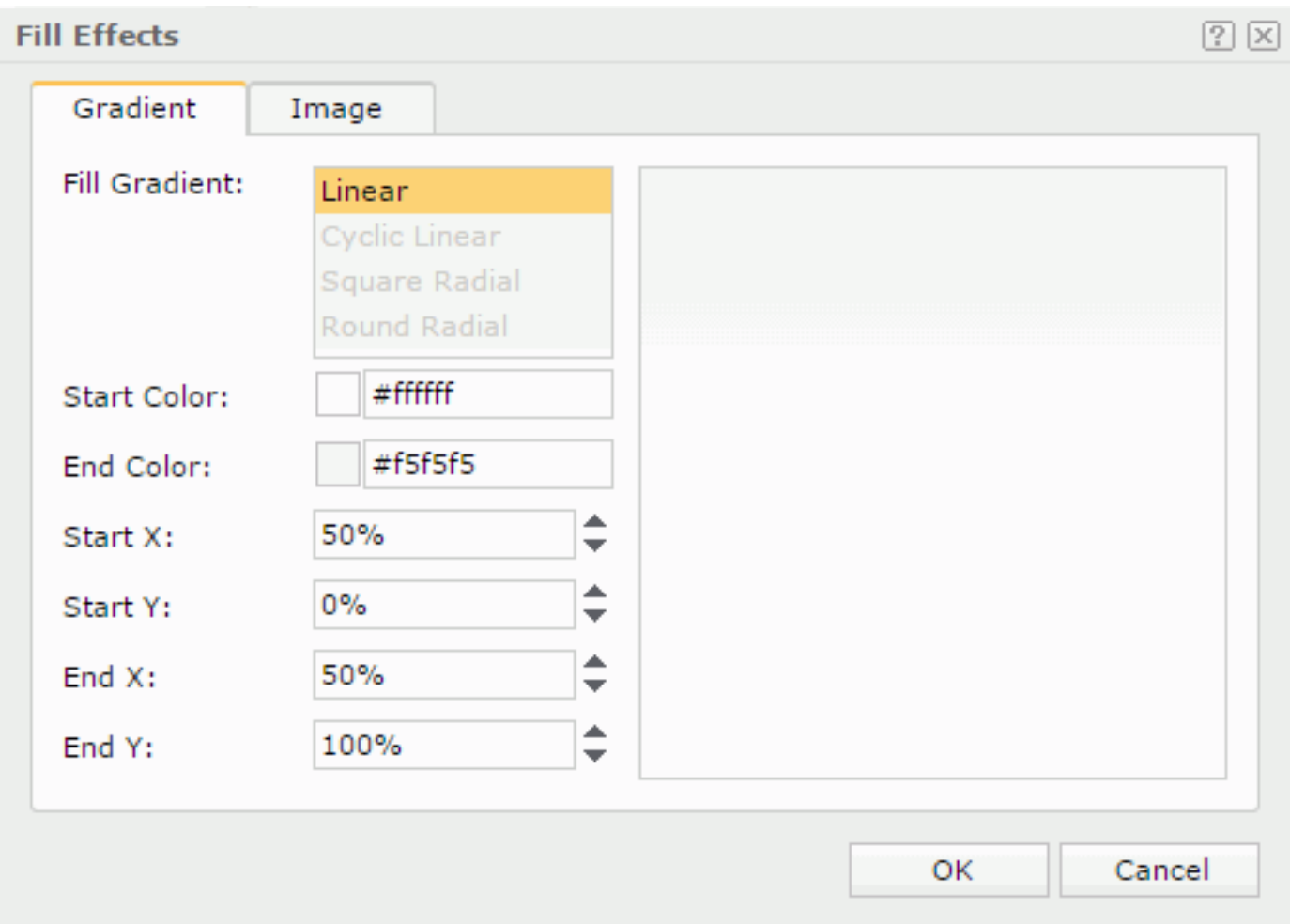
Does not retain any changes and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Gradient tab

This tab allows you to specify the gradient colors to fill the object.



Fill Gradient

Displays the type of gradient to fill the object. It is Linear by default.

Start Color

Specifies a color in the gradient graph.

End Color

Specifies the other color in the gradient graph.

Start X

Specifies the horizontal position, measured in the percentage of the object's width, from the left edge of the object, where the gradient starts.

Start Y

Specifies the vertical position, measured in the percentage of the object's height, from the top edge of the object, where the gradient starts.

End X

Specifies the horizontal position, measured in the percentage of the object's width, from the left edge of the object, where the gradient ends.

End Y

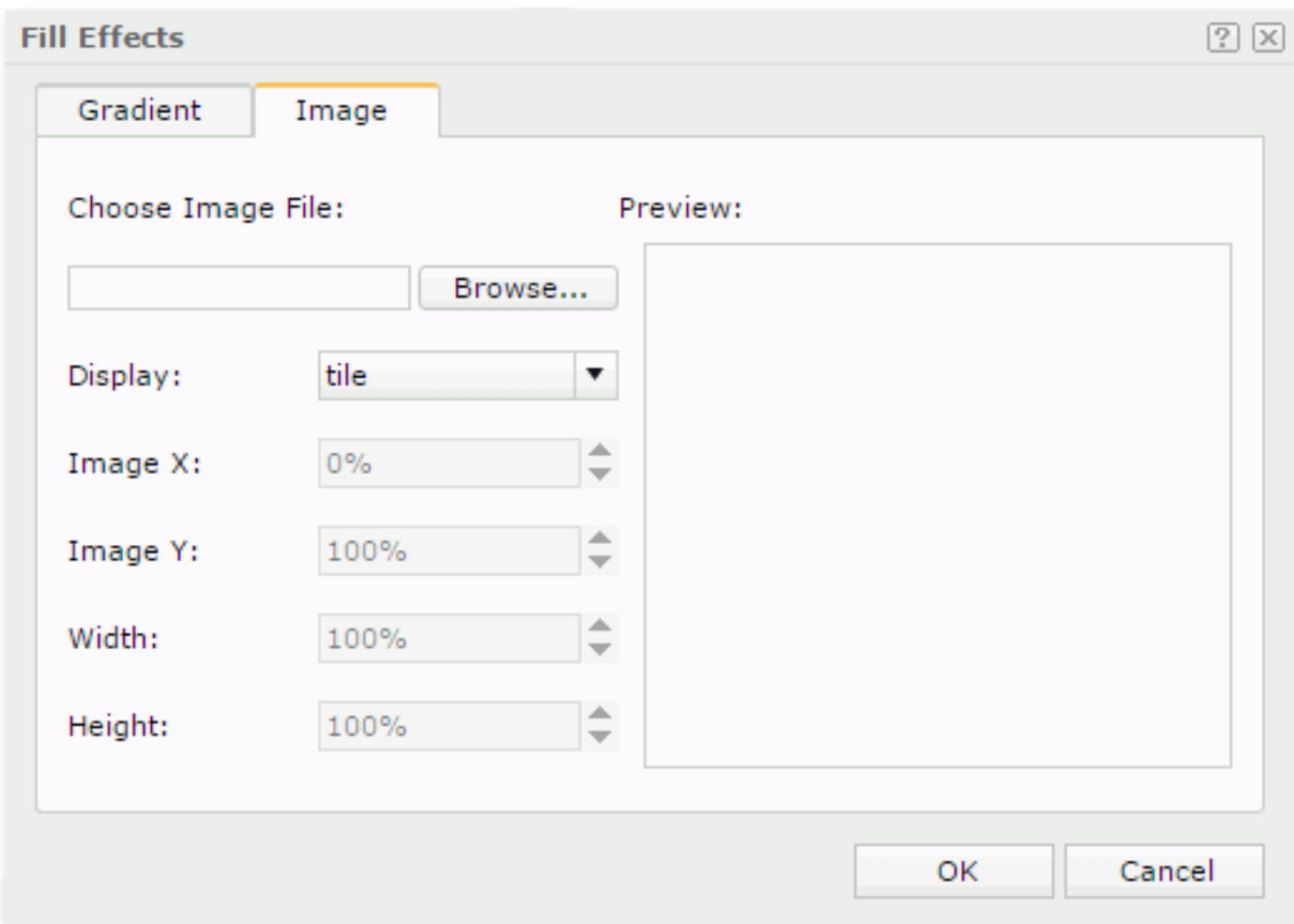
Specifies the vertical position, measured in the percentage of the object's height, from the top edge of the object, where the gradient ends.

Preview

Displays a preview of your settings.

Image tab

This tab allows you to specify an image to fill the object.



Choose Image File

Specifies the image file. Click Browse to select the required image.

Display

Specifies the layout style of the image in the object. It can be one of the following: Tile, Center and Scaled.

Image X

Displays the left position of the area, measured in the percentage of the image's width, from the left edge of the image.

Image Y

Displays the bottom position of the area, measured in the percentage of the image's width, from the bottom edge of the image.

Width

Displays the width of the area, measured in the percentage of the image's width.

Height

Displays the height of the area, measured in the percentage of the image's height.

The above four options, namely Image X, Image Y, Width and Height, are used to define an area in the image and put the defined area of the image into the object. Their values are fixed.

Preview

Displays a preview of your settings.

Filter Control Properties dialog

This dialog appears when you right-click a filter control and select Properties from the shortcut menu. It helps you to edit the properties of the filter control and contains the following tabs:

- General tab
- Font tab
- Border tab
- Title tab

OK

Applies the settings and closes this dialog.

Cancel

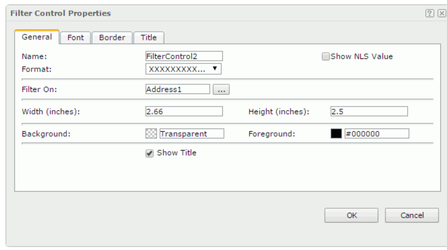
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the filter control.



Name

Specifies the name of the filter control.

Show NLS Value

Specifies to show the translated name of the display name of the filter control in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the filter control is not modified.

Format

Specifies the field value format in the filter control.

Filter On

Displays the fields that the filter control is based on. You can click the button ... to open the Select Field dialog to select other fields.

Width

Specifies the width of the filter control.

Height

Specifies the height of the filter control.

Background

Specifies the background color of the filter control.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Foreground

Specifies the foreground color of the filter control.

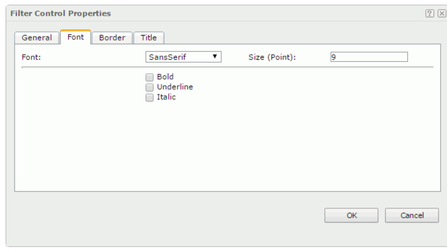
To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB.

Show Title

Specifies whether to show the title of the filter control.

Font tab

This tab shows the font-related information of the text in the filter control.



Font

Specifies the font face of the text.

Size

Specifies the font size of the text.

Bold

Specifies whether to make the text bold or not.

Underline

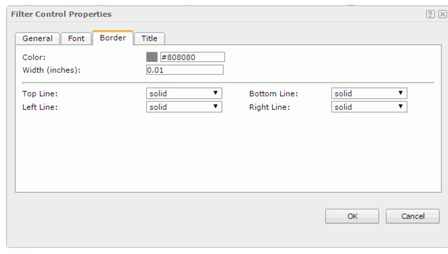
Specifies whether the text will be underlined or not.

Italic

Specifies whether to make the text italic or not.

Border tab

This tab shows information about borders of the filter control.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

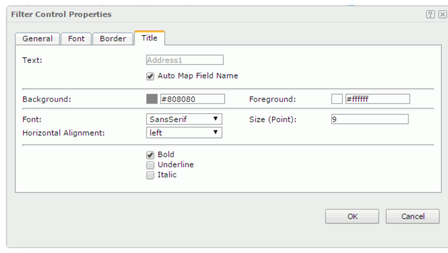
Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Title tab

This tab shows information about the title of the filter control. It is available when the Show Title option is selected in the General tab of the same dialog.



Text

Specifies the text of the title.

- **Auto Map Field Name**
  Specifies whether to use the field name as the title. If selected, the Text box above is disabled for inputting.

Background

Specifies the background color of the title.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Foreground

Specifies the foreground color of the title.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB.

Font

Specifies the font face of the text.

Size

Specifies the font size of the text.

Horizontal Alignment

Specifies the horizontal alignment mode of the text.

Bold

Specifies whether to make the text bold or not.

Underline

Specifies whether the text will be underlined or not.

Italic

Specifies whether to make the text italic or not.

Filter dialog

The dialog appears when you click Menu > Edit > Filter, or the Filter button on the Standard toolbar. It helps you to set criteria for filtering records, and has the following two modes:

- Basic
- Advanced

Inspector

Opens the Filter Inspector dialog.

Advanced/Basic

Switches the dialog to the advanced/basic mode.

OK

Closes the dialog and filters the records of the report with the specified criteria.

Cancel

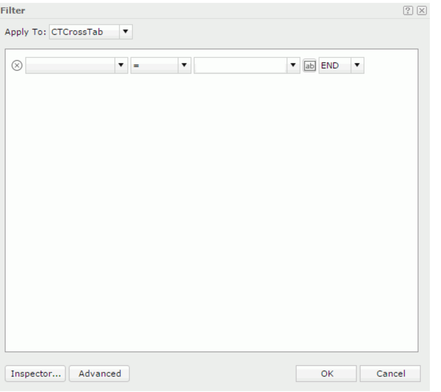
Cancels to set criteria for filtering records and exits the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Basic mode

The basic mode provides function for creating simple filter conditions which are connected by AND and OR operators.



Apply to

Specifies the component to which the filter will be applied.

Deletes the current condition line.

Field

Specifies the field to be filtered.

Operator

Specifies the operator to compose the filter expression.

- **=**
  Equal to
- **>**
  Greater than
- **>=**
  Greater than or equal to
- **<**
  Less than
- **<=**
  Less than or equal to
- **!=**
  Not equal to
- **[not] in**
  Causes an enumerated list of values to appear in the WHERE clause predicate, used for evaluating for a true condition.
- **[not] like**
  Like string pattern matching operator is used to compare the first expression string value to the pattern string (the second expression). If you want to use wildcard character in the pattern string, only "_" and "%" are supported.
- **[not] between**
  Allows the system to evaluate whether or not data values are located between a range of values indicated in the predicate.
- **is [not] null**
  It is used in WHERE clause predicates to match null values occurring in a specified data field.

Value

Specifies the value of how to filter the field. You can enter the value manually in the text box, select a value from the drop-down list, or click the button and then select a parameter from the drop-down list as the value. When you type in the value manually, if multiple values are required, they should be separated with ",", and if "," or "\" is contained in the values, write it as "\," or "\\".

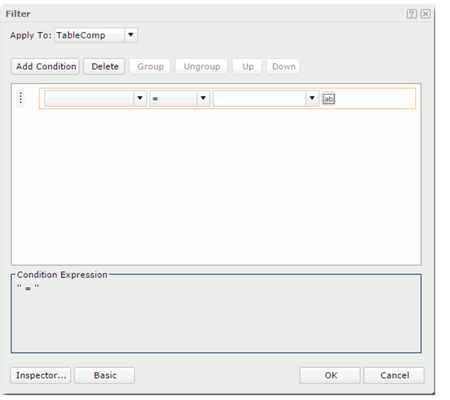
Logic

Lists the logic operator.

- **AND**
  Logic operator AND which is applied to this and the next line.
- **OR**
  Logic operator OR which is applied to this and the next line.
- **END**
  Logic operator END which is applied to this line.

Advanced mode

The advanced mode enables the building of more complex filter conditions via the grouping of conditions.



Apply to

Specifies the component to which the filter will be applied.

Add Condition

Adds a new condition line.

Delete

Deletes the selected condition line or group.

Group

Makes the selected condition lines in a group. Condition lines can also be added to an existing group by selecting the condition lines and the group while holding the Ctrl button, and then clicking the Group button.

Ungroup

Makes the selected condition line or group ungrouped.

Up

Moves the selected condition line or group up to a higher level.

Down

Moves the selected condition line or group down to a lower level.

Logic

Specifies the logic operator.

- **AND**
  Logic operator AND which is applied to this and the next line.
- **OR**
  Logic operator OR which is applied to this and the next line
- **AND NOT**
  Logic operator AND NOT which is applied to this and the next line.
- **OR NOT**
  Logic operator OR NOT which is applied to this and the next line.

Field

Specifies the field to be filtered.

Operator

Specifies the operator to compose the filter expression.

Value

Specifies the value of how to filter the field. You can enter the value manually in the text box, select a value from the drop-down list, or click the button and then select a parameter from the drop-down list as the value. When you type in the value manually, if multiple values are required, they should be separated with ",", and if "," or "\" is contained in the values, write it as "\," or "\\".

Condition Expression

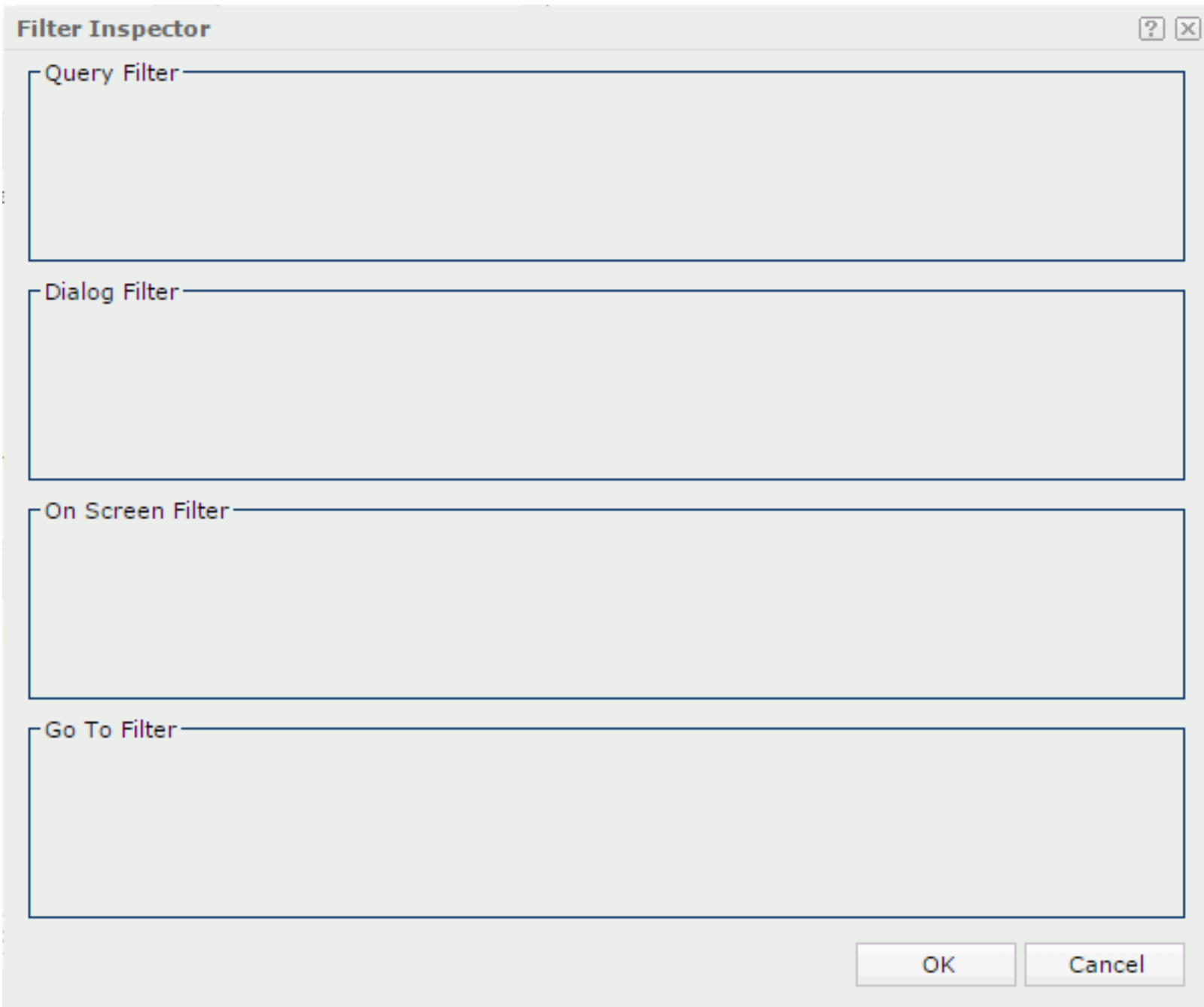
Displays the SQL statement of the filter.

Related topics:

- Filtering the data components in a report

Filter Inspector dialog

The dialog appears when you click the Inspector button in the Filter dialog. It lists all the filters the current web report is using for your information.



Query Filter

Lists the filters created via the Query Filter dialog.

Dialog Filter

Lists the filters created via the Filter dialog.

On Screen Filter

Lists the filters created via the Filter panel and via filter controls.

Go To Filter

Lists all the go to filters that are created when performing the go-to-by-value action on the report.

OK

Applies the changes and closes the dialog.

Cancel

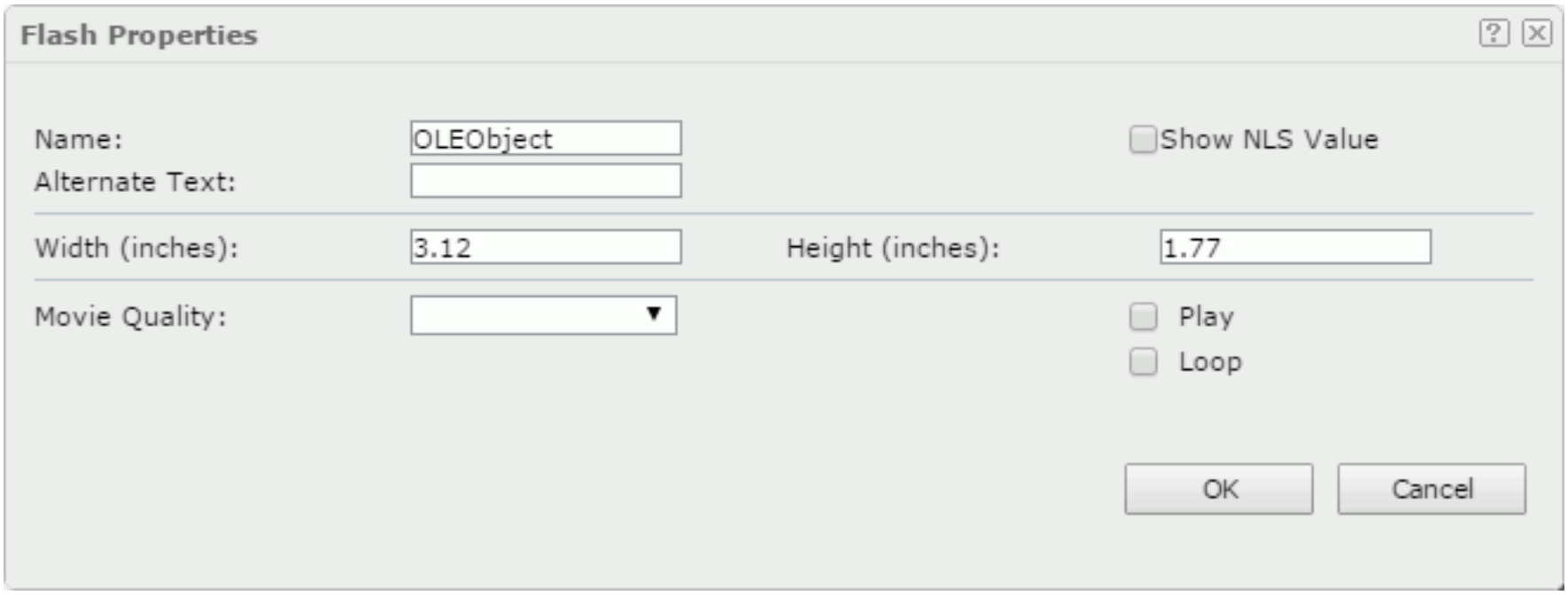
Cancels the changes and exits the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Flash Properties dialog

The dialog appears when you right-click a flash object and select Properties from the shortcut menu. It helps you to edit the properties of the flash object.



Name

Specifies the display name of the flash, which will be shown on the shortcut menu of the flash.

Show NLS Value

Specifies to show the translated name of the display name of the flash in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the flash is not modified.

Alternate Text

Specifies the alternate text which will be shown if the flash cannot be displayed.

Width

Specifies the width of the flash.

Height

Specifies the height of the flash.

Movie Quality

Specifies the quality of the flash.

Play

Specifies whether to start the flash file automatically when the report is opened.

Loop

Specifies whether to play the flash file repeatedly.

OK

Applies the settings and closes this dialog.

Cancel

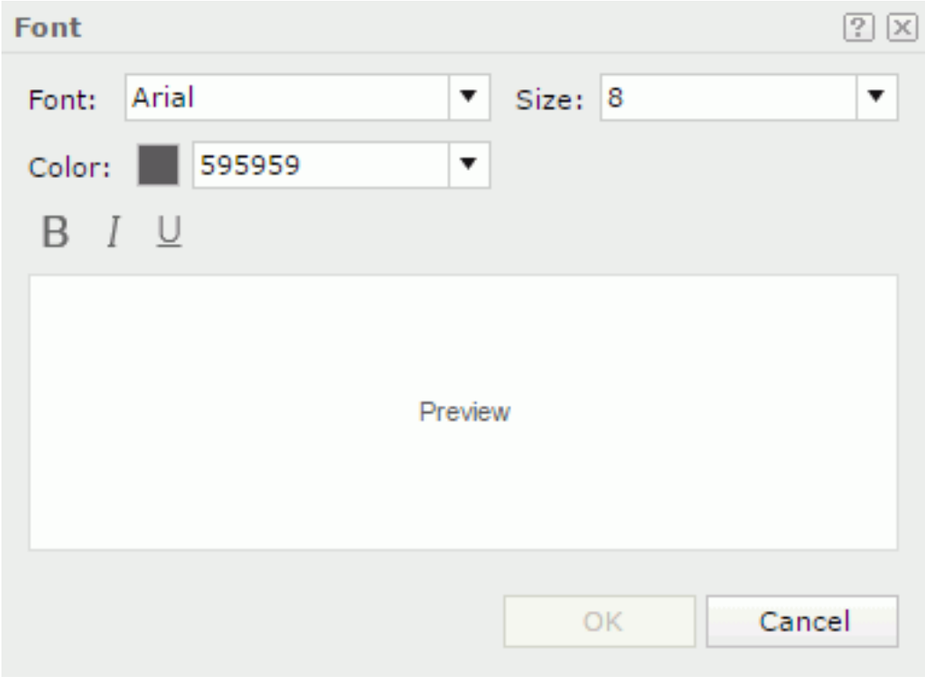
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Font dialog

This dialog appears when you select a label or field and click the Font button A on the toolbar. It helps you to specify the font format of the label or field.



Font

Specifies the font face.

Size

Specifies the font size.

Color

Specifies the font color.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB.

B

Specifies whether to make the object bold.

I

Specifies whether to make the object italic.

U

Specifies whether to underline the object.

Preview

Shows the font effect.

OK

Applies the font format you specified and closes this dialog.

Cancel

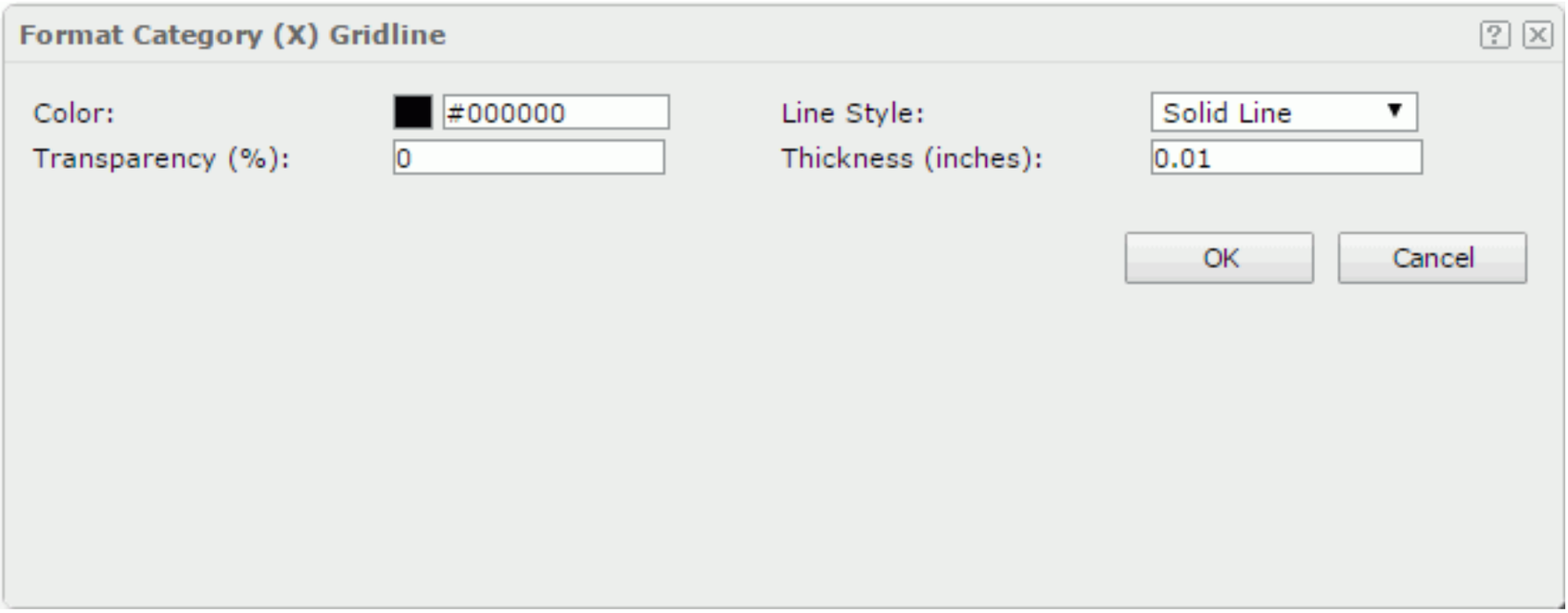
Cancels to change the font format and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Format Category (X) Gridline dialog

This dialog appears when you right-click a chart and then select Format Gridlines (unavailable to pie, indicator, heat map and org charts) > Format Category (X) Gridline from the shortcut menu. It helps you to format the gridlines shown on the category (X) axis.



Color

Specifies the color schema of the gridlines.

Transparency

Specifies the transparency of the color that is applied to the gridlines.

Line Style

Specifies the line style of value gridlines.

Thickness

Specifies the thickness of value gridlines.

OK

Applies the changes and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Format Graph dialog

This dialog appears when you right-click an indicator chart and select Format Graph from the shortcut menu. This dialog helps you to format the indicators of the indicator chart and contains the following tabs:

- General tab
- Border tab

OK

Applies the settings and closes this dialog.

Cancel

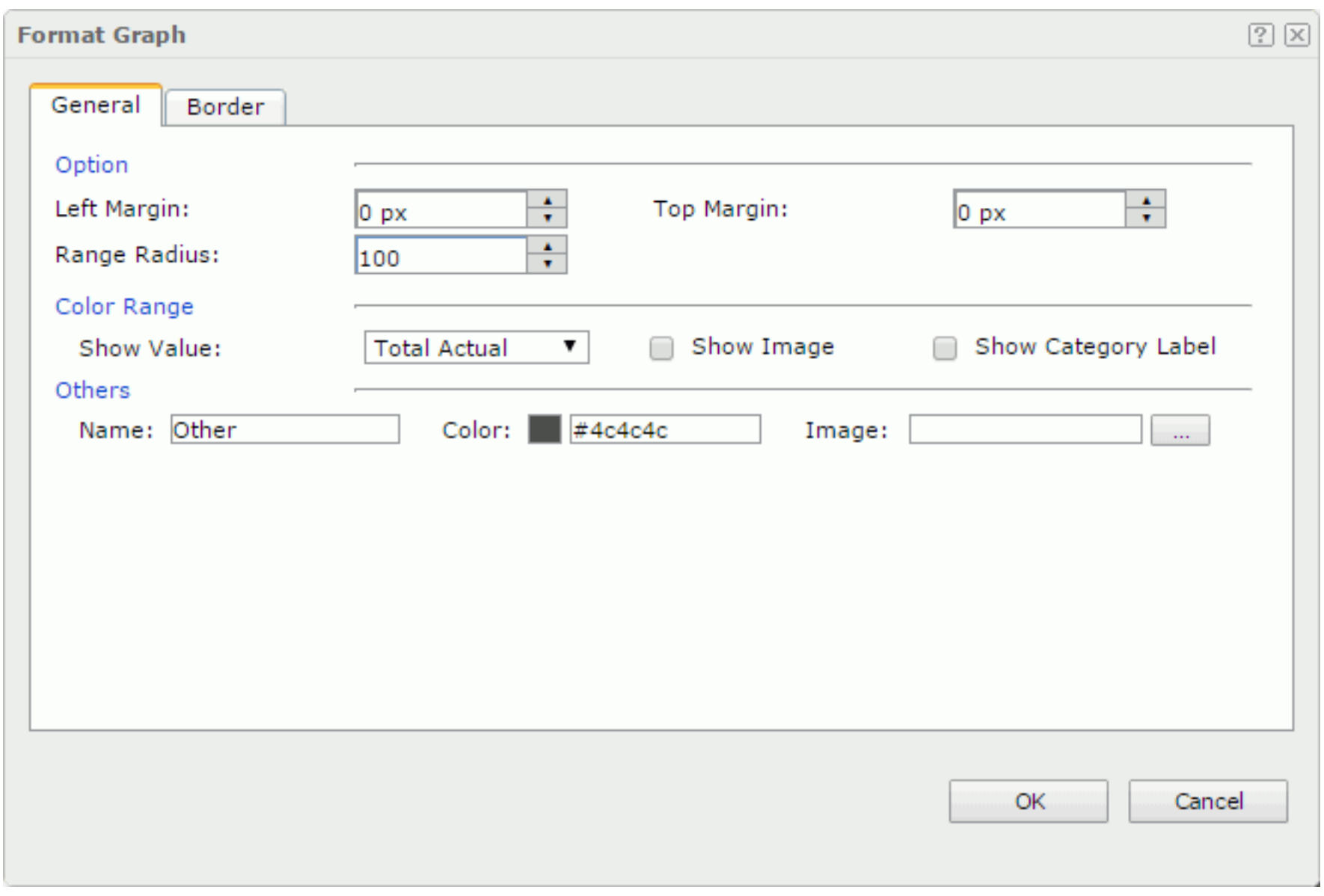
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

Specifies the general properties of the indicator chart.



Option

Specifies the options for the indicators.

- **Left Margin**
  Specifies the gap between the left labels and the left indicators, in pixels.
- **Top Margin**
  Specifies the gap between the top labels and the top indicators, in pixels.
- **Range Radius**
  Specifies the relative size of an indicator in a percentage of the total indicator size.

Color Range

Specifies different colors or images to fill the indicators.

- **Show Value**
  Specifies the value you want to format.
- **Show Image**
  Specifies to use images to represent the value ranges.
- **Show Category Label**
  Specifies whether to show the category labels for the indicators. The label displays the category value each indicator represents.

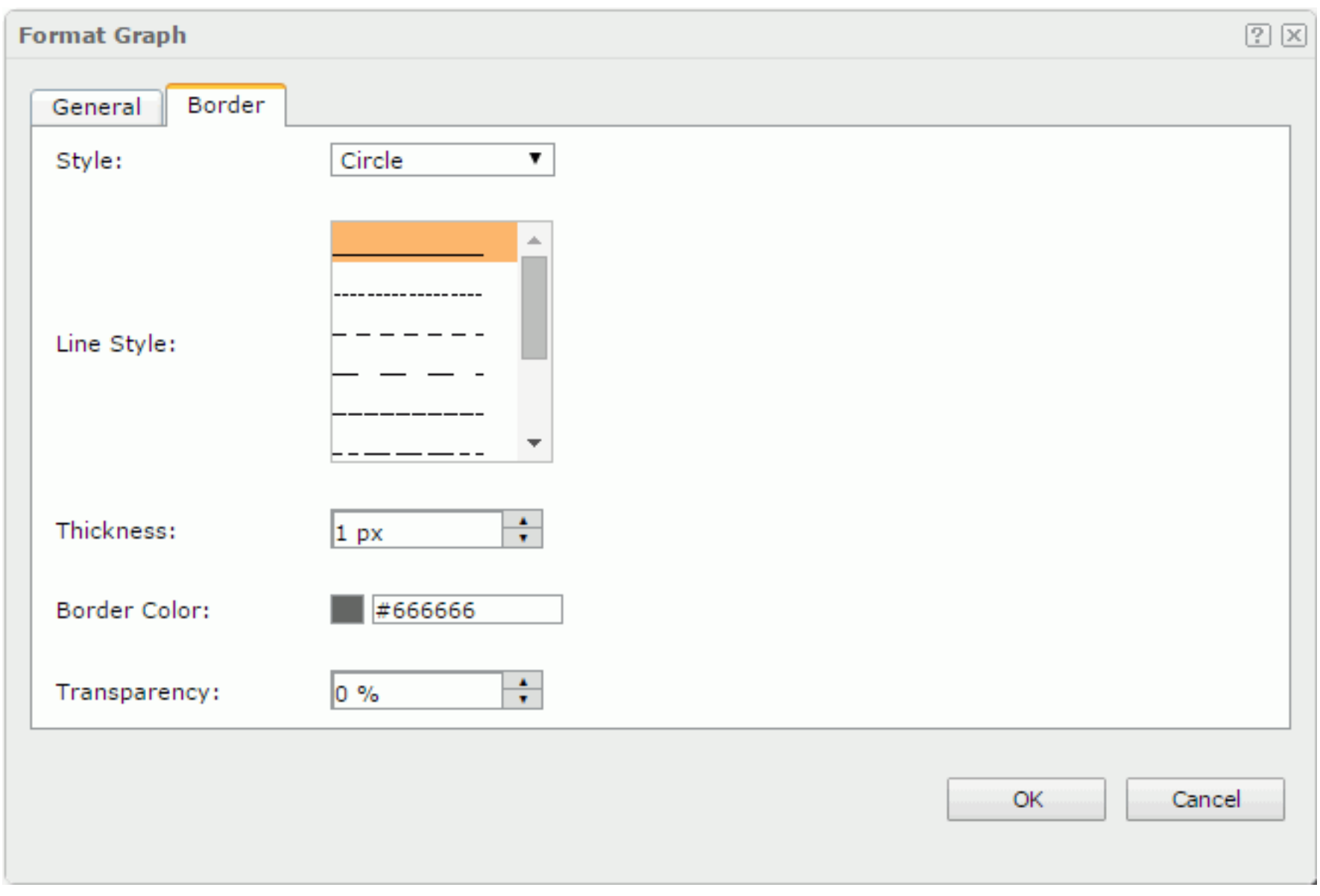
Others

Specifies the properties for values that do not fall into any of the ranges you have defined. Available only when the fields displayed on the value axis of the indicator chart are of String or Numeric type.

- **Name**
  Specifies the name of the values.
- **Color**
  Specifies the color of the values.
- **Image**
  Specifies the image to represent the values.

Border tab

Specifies the properties for border of the indicators. Applied when the indicators are not represented by images.



Style

Specifies the style of the indicator.

Line Style

Specifies the line style to apply to the border.

Thickness

Specifies the thickness of the border, in pixels.

Border Color

Specifies the color of the border.

Transparency

Specifies the transparency for color of the border.

Format Legend dialog

This dialog appears when you right-click a chart and select Format Legend (unavailable to org chart) from the shortcut menu. This dialog helps you to format the legend of the chart and contains the following tabs:

- General tab
- Placement tab
- Border tab
- Font tab

OK

Applies the settings and closes this dialog.

Cancel

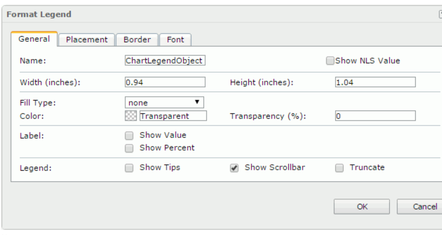
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the chart legend.



Name

Specifies the display name of the chart legend, which will be shown on the shortcut menu of the legend.

Show NLS Value

Specifies to show the translated name of the display name of the chart legend in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the chart legend is not modified.

Width

Specifies the width of the chart legend.

Height

Specifies the height of the chart legend.

Fill Type

Specifies the type for filling the chart legend.

Color

Indicates the background color of the chart legend.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Transparency

Specifies the transparency of the legend background.

Show Value

Specifies whether to show the value of each legend.

Show Percent

Specifies whether to show the percentage of each legend.

Show Tips

Specifies whether to show the corresponding data information when the mouse pointer points at a target in the chart legend.

Show Scrollbar

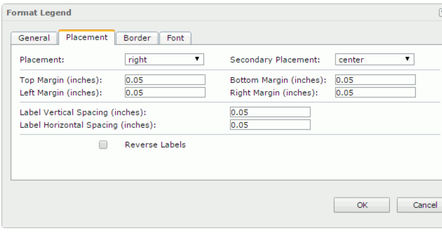
Specifies whether to show a scrollbar on the legend to fully view the legend content when the content does not fit into the legend.

Truncate

Specifies whether to truncate the legend entry label text when the text overflow the labels.

Placement tab

This tab shows the position-related information of the chart legend.



Placement

Specifies the position of the legend in the platform.

Secondary Placement

Specifies the position of the legend on the basis of the Placement property.

Top Margin

Specifies the distance between the legend labels and the top border of the legend.

Bottom Margin

Specifies the distance between the legend labels and the bottom border of the legend.

Left Margin

Specifies the distance between the legend labels and the left border of the legend.

Right Margin

Specifies the distance between the legend labels and the right border of the legend.

Label Vertical Spacing

Specifies the vertical distance between two adjacent legend labels.

Label Horizontal Spacing

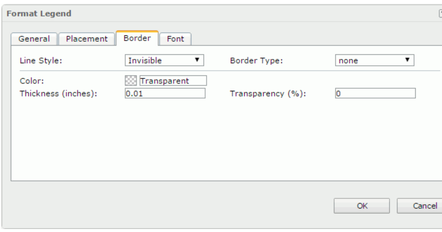
Specifies the horizontal distance between two adjacent legend labels.

Reverse Labels

Specifies whether or not to arrange the legend labels in a reverse order.

Border tab

This tab shows information about borders of the chart legend.



Line Style

Specifies the line style of the legend borders.

Border Type

Specifies the type of the legend borders.

Color

Specifies the color of the legend borders.

Transparency

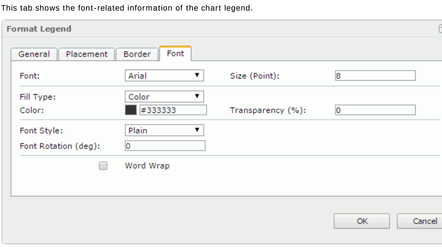
Specifies the transparency of the legend borders.

Thickness

Specifies the thickness of the legend borders.

Font tab

This tab shows the font-related information of the chart legend.



Font

Specifies the font face for the legend labels.

Size

Specifies the font size for the legend labels.

Fill Type

Specifies the fill type for the legend labels.

Color

Specifies the color for the legend labels.

Transparency

Specifies the transparency for the legend labels, in percent.

Font Style

Specifies the font style of the text. It can be one of the following: Plain, Bold, Italic and Bold Italic.

Font Rotation

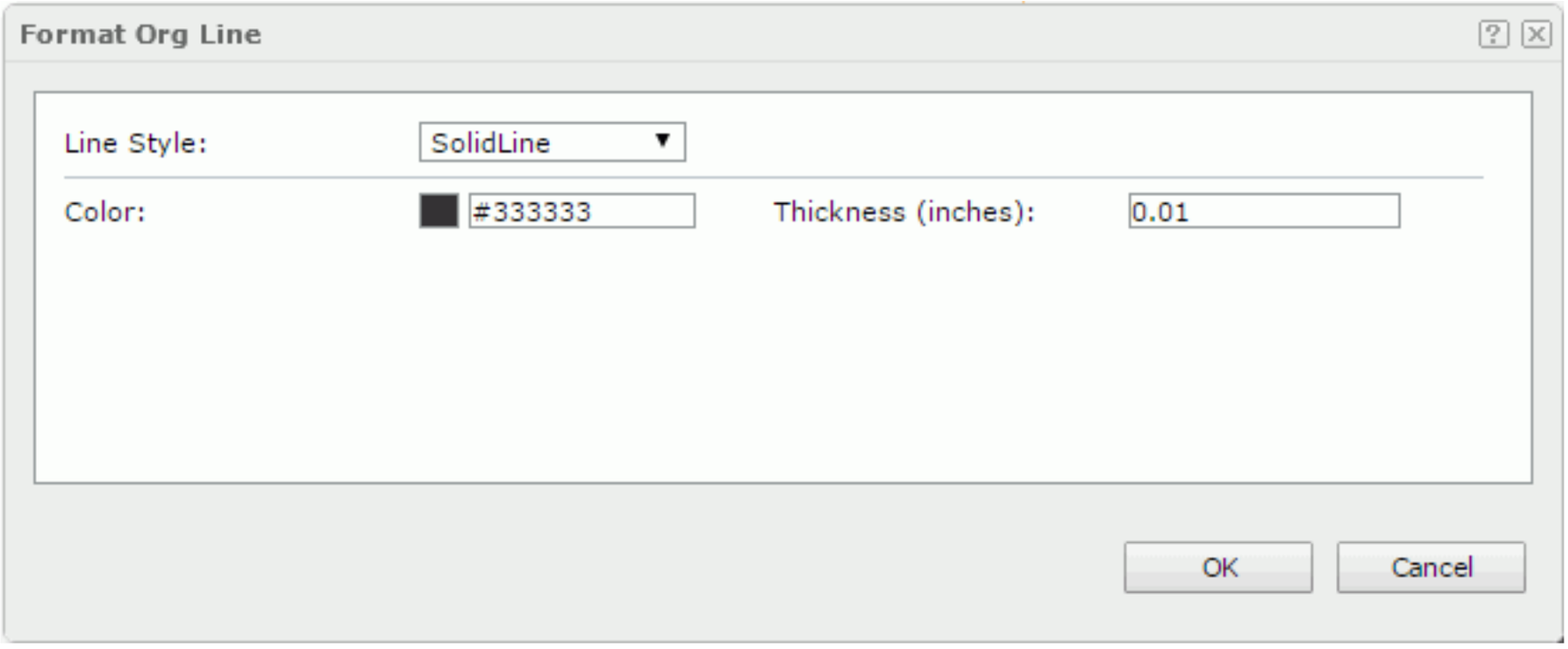
Specifies the rotation angle of each legend label around its center, in degrees.

Word Wrap

Specifies whether or not to enable word wrapping for the label text.

Format Org Line dialog

This dialog appears when you right-click an org chart and select Format Line from the shortcut menu. It helps you to format the lines of the org chart.



Line Style

Specifies the style of the lines. Choose a style from the drop-down list.

Color

Specifies the color of the line borders.

Thickness

Specifies the thickness of the line borders.

OK

Applies the settings and closes this dialog.

Cancel

Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Format Org Node dialog

This dialog appears when you right-click an org chart and select Format Node from the shortcut menu. It helps you to format the nodes of the org chart and contains the following tabs:

- General tab
- Border tab

OK

Applies the settings and closes this dialog.

Cancel

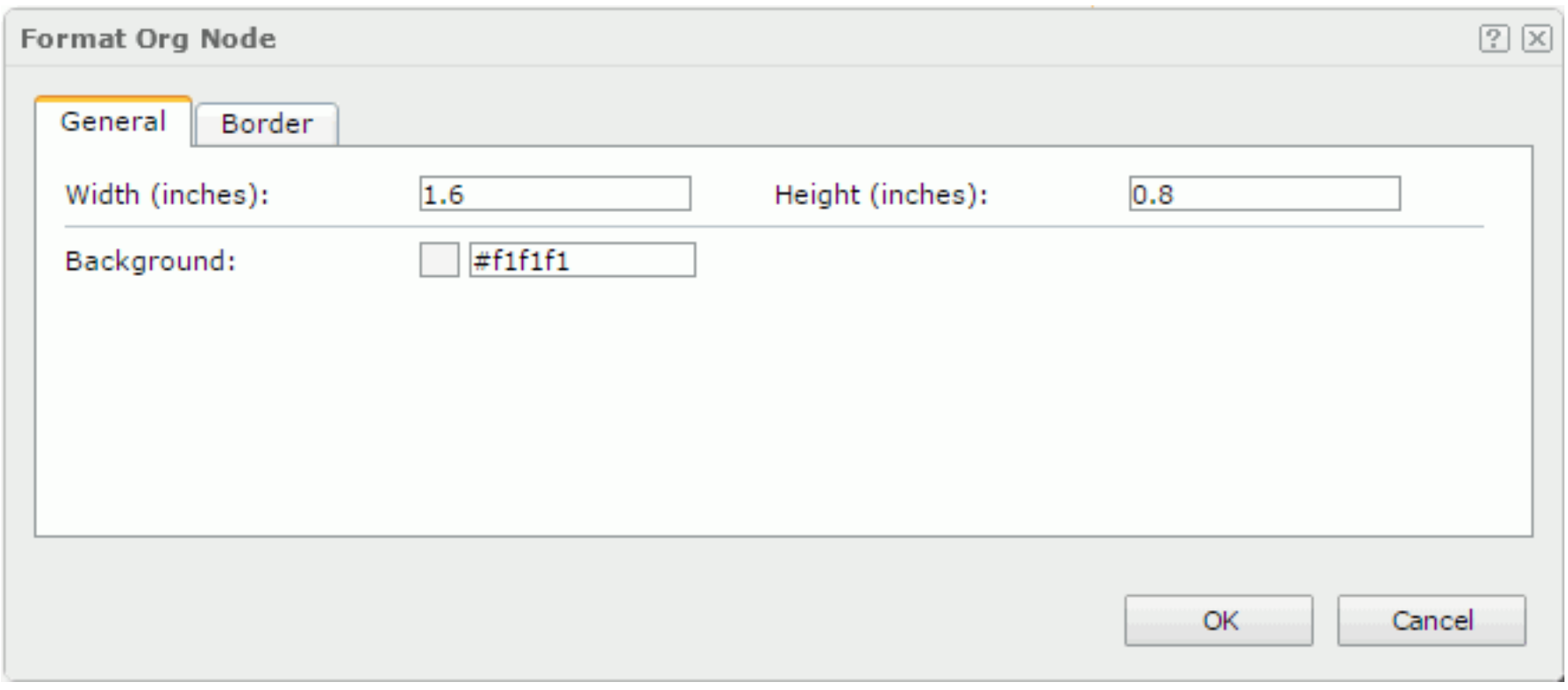
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the org chart nodes.



Width

Specifies the width of the nodes in inches.

Height

Specifies the height of the nodes in inches.

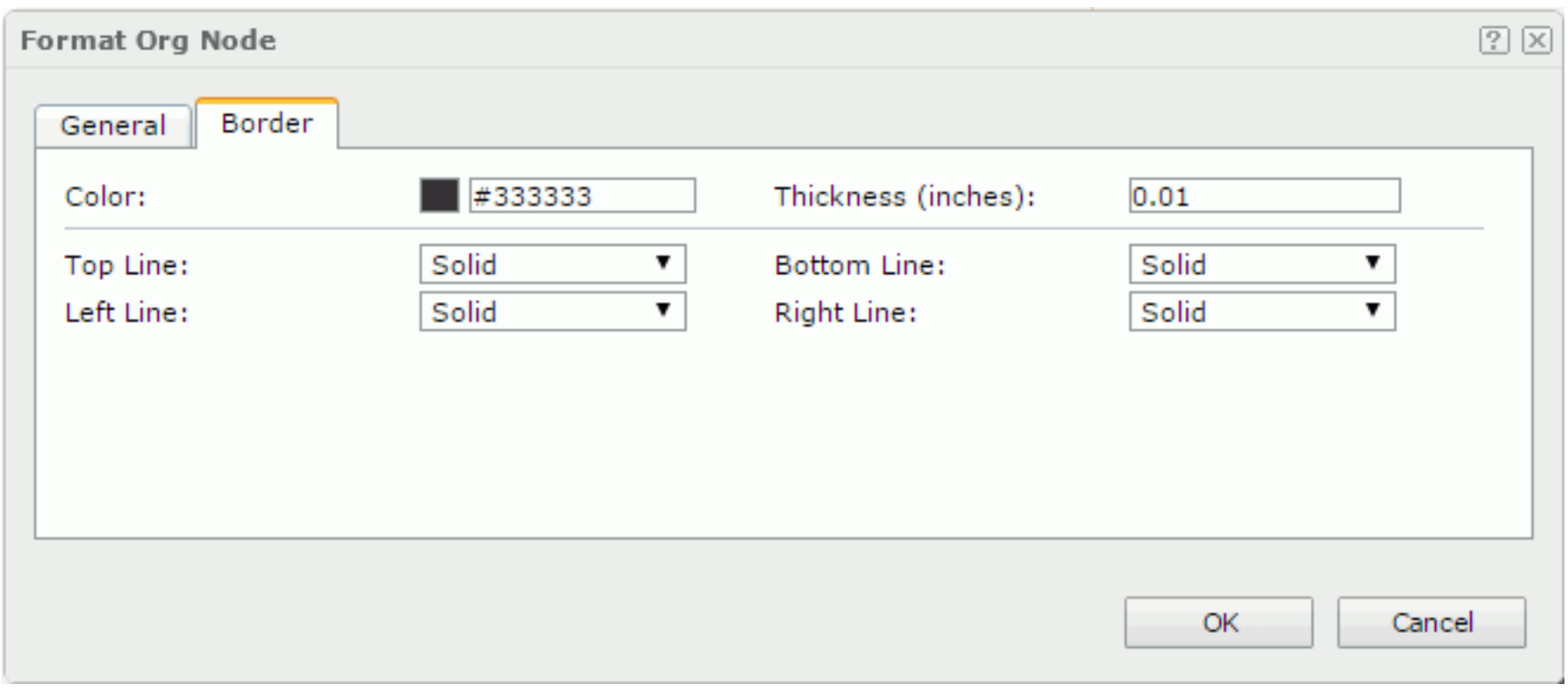
Background

Indicates the background color of the nodes.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the org chart nodes.



Color

Specifies the color of the node borders.

Thickness

Specifies the thickness of the node borders.

Top Line Style

Specifies the line style of the top border. Choose a style from the drop-down list.

Bottom Line Style

Specifies the line style of the bottom border. Choose a style from the drop-down list.

Left Line Style

Specifies the line style of the left border. Choose a style from the drop-down list.

Right Line Style

Specifies the line style of the right border. Choose a style from the drop-down list.

Format Paper dialog

This dialog appears when you right-click a chart and select Format Paper from the shortcut menu. It helps you to format the paper of the chart and contains the following tabs:

- General tab
- Border tab
- Coordinate tab
- Graph tab
- Threshold Line tab

OK

Applies the settings and closes this dialog.

Cancel

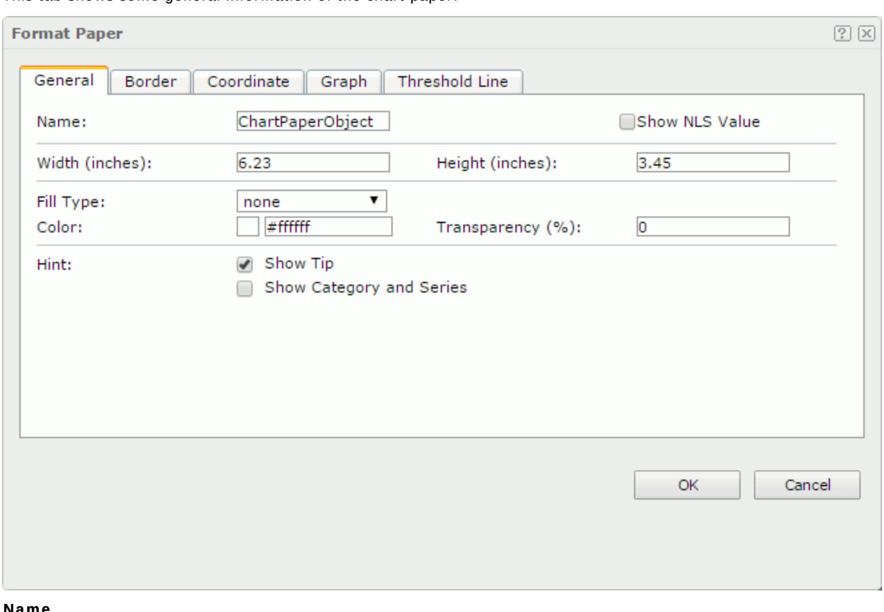
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the chart paper.



Name

Specifies the display name of the chart paper, which will be shown on the shortcut menu of the paper.

Show NLS Value

Specifies to show the translated name of the display name of the chart paper in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the chart paper is not modified.

Width

Specifies the width of the chart paper.

Height

Specifies the height of the chart paper.

Fill Type

Specifies the type for filling the chart paper.

Color

Indicates the background color of the chart paper.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Transparency

Specifies the transparency of the chart paper background.

Show Tip

Specifies whether to show the corresponding data information when the mouse pointer points at a target in the chart paper. Unavailable to org chart.

Show Category and Series

Specifies whether to show data of the category and value axes when the mouse pointer points at a target in the chart paper. It takes effect only when the option Show Tips is checked. Unavailable to org chart.

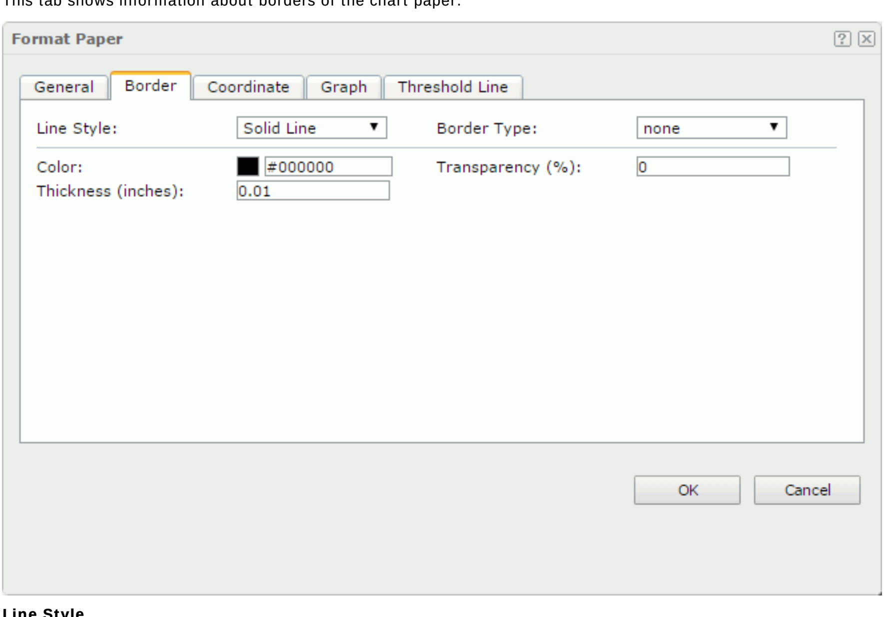
Layout Mode

Specifies the layout mode of the org chart. Available to org chart only.

- **Top Down**
  The org chart tree expands from top to bottom.
- **Bottom Up**
  The org chart tree expands from bottom to top. It is the reverse of the top down type.
- **Left Right**
  The org chart tree expands from left to right.

Border tab

This tab shows information about borders of the chart paper.



Line Style

Specifies the line style of the chart paper borders.

Border Type

Specifies the type of the chart paper borders.

Color

Specifies the color of the chart paper borders.

Transparency

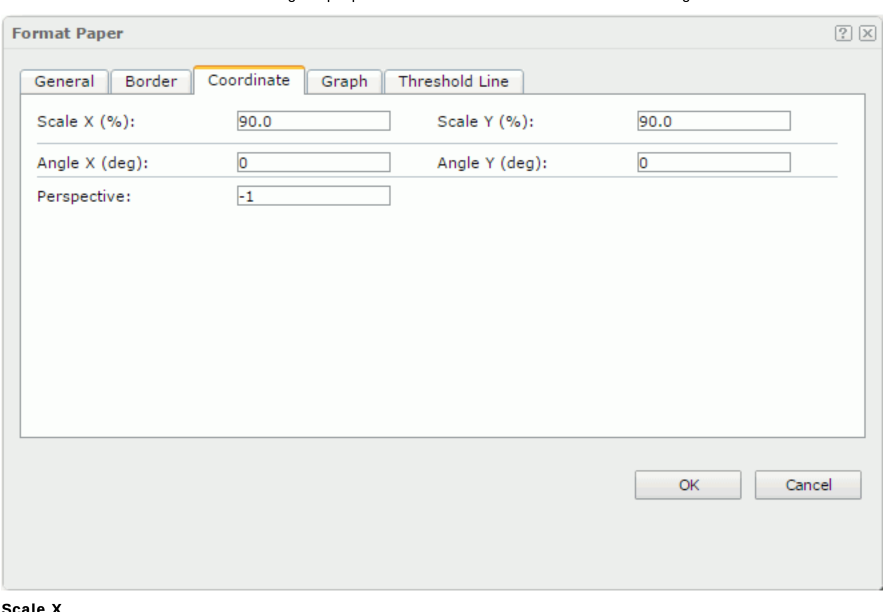
Specifies the transparency of the chart paper borders.

Thickness

Specifies the thickness of the chart paper borders.

Coordinate tab

You can use this tab to view and configure properties of the coordinates. Unavailable to org chart.



Scale X

Specifies the scaling ratio for the X axis.

Scale Y

Specifies the scaling ratio for the Y axis.

Angle X

Specifies the rotation angle around the X axis. Applies to 3-D charts and pie charts.

Angle Y

Specifies the rotation angle around the Y axis. Applies to 3-D charts and pie charts.

Perspective

Specifies the perspective effect of the chart. Input an integer to set the effect.

Graph tab

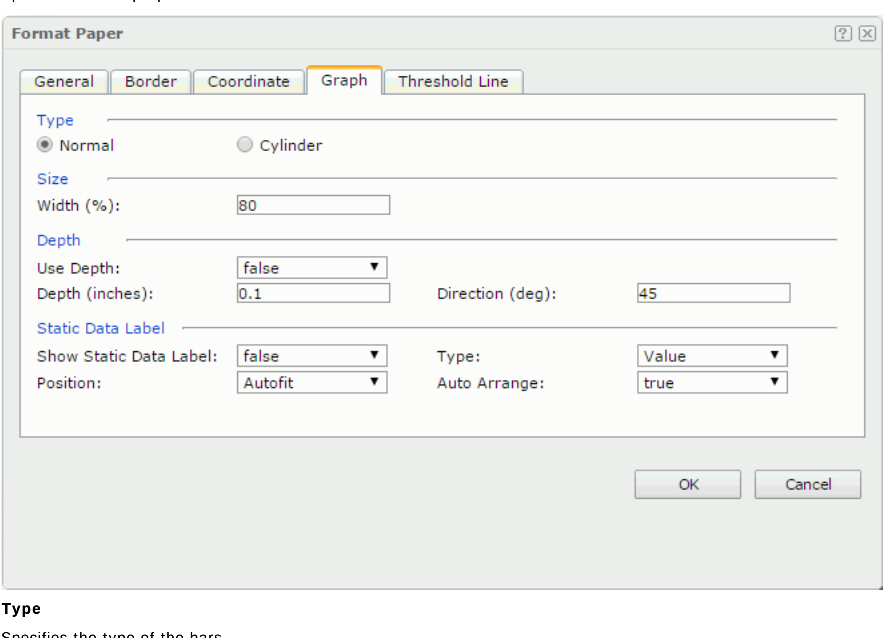
You can use this tab to view and configure properties of the graphic object. Unavailable to org chart.

This tab differs according to the following chart types:

- Bar/Bench
- Line/Area
- Pie/Donut

Bar/Bench

Specifies the bar properties.



Type

Specifies the type of the bars.

- **Normal**
  Specifies to make the bars be quadrate.
- **Cylinder**
  Specifies to make the bars be columned.

Size

Specifies the size of the bars.

- **Width**
  Specifies the width of the bars.

Depth

Specifies the depth properties for the bars.

- **Use Depth**
  Specifies whether to make the bars visually three-dimensional.
- **Depth**
  Specifies the depth of the bars.
- **Direction**
  Specifies the angle of the axis along the depth of the bars.

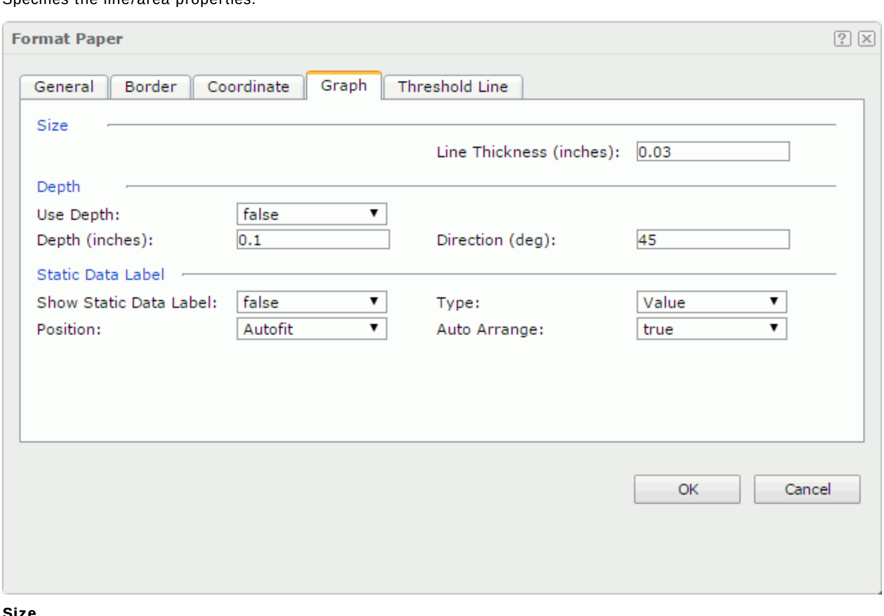
Static Data Label

Specifies properties of the static data labels.

- **Show Static Data Label**
  Specifies whether or not to show the static data labels on the bars. Only when it is checked can the following static data label related properties take effect.
- **Type**
  Specifies in which way the static data labels will be displayed.
  - **Value** - Shows the value for each bar.
  - **Percent** - Shows the percentage of each bar to the total.
  - **Value Percent** - Shows the value and the percentage for the each bar.
- **Position**
  Specifies the position of the static data labels on the bars.
  - **Autofit**
    If selected, the static data labels will be displayed automatically.
  - **Outside Top**
    If selected, the static data labels will be displayed on the outside top of the bars.
  - **Inside Top**
    If selected, the static data labels will be displayed on the inside top of the bars.
  - **Inside Center**
    If selected, the static data labels will be displayed at the inside center of the bars.
  - **Inside Bottom**
    If selected, the static data labels will be displayed at the inside bottom of the bars.
- **Auto Arrange**
  Specifies whether to display the static data labels inside the bars at the best position. Available only when the static data label's position is set to one of the following: Inside Center, Inside Top and Inside Bottom.
  - **True**
    The static data label will be displayed horizontally at the specified position if the bar has enough room horizontally, otherwise, it will be displayed vertically. If a bar does not have enough room both vertically and horizontally, its static data label will not be displayed.
  - **False**
    The static data labels will be displayed at the specified position, and if the labels get overlapping, some of them will not be displayed.

Line/Area

Specifies the line/area properties.



Size

Specifies the size of the lines. Available to line chart only.

- **Line Thickness**
  Specifies the thickness of the lines.

Depth

Specifies the depth properties of the lines/areas.

- **Use Depth**
  Specifies whether to make the lines/areas visually three-dimensional.
- **Depth**
  Specifies the depth of the lines/areas.
- **Direction**
  Specifies the angle of the axis along the depth of the lines/areas.

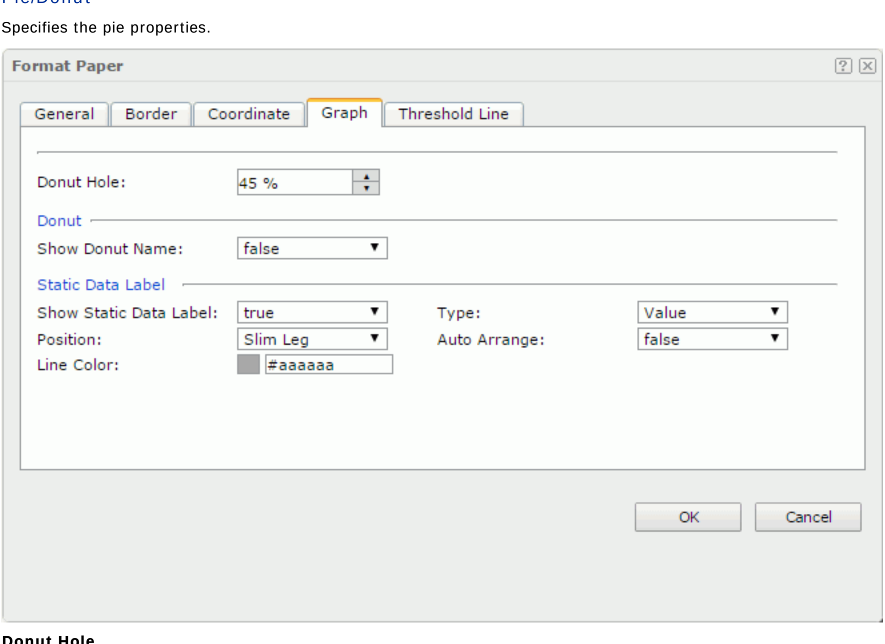
Static Data Label

Specifies properties of the static data labels.

- **Show Static Data Label**
  Specifies whether or not to show the static data labels on the lines/areas. Only when it is checked can the following static data label related properties take effect.
- **Type**
  Specifies in which way the static data labels will be displayed.
  - **Value** - Shows the value for each line/area node.
  - **Percent** - Shows the percentage of each line/area node to the total.
  - **Value Percent** - Shows the value and the percentage for each line/area node.
- **Position**
  Specifies the position of the static data labels on the lines/areas.
  - **Autofit**
    If selected, the static data labels will be displayed automatically.
  - **Top Center**
    If selected, the static data labels will be displayed on the top center of the line/area nodes.
  - **Top Left**
    If selected, the static data labels will be displayed on the top left of the line/area nodes.
  - **Top Right**
    If selected, the static data labels will be displayed on the top right of the line/area nodes.
  - **Bottom Left**
    If selected, the static data labels will be displayed at the bottom left of the line/area nodes.
  - **Bottom Center**
    If selected, the static data labels will be displayed at the bottom center of the line/area nodes.
  - **Bottom Right**
    If selected, the static data labels will be displayed at the bottom right of the line/area nodes.
- **Auto Arrange**
  Specifies whether to display the static data labels at the best position. Not supported on line and area charts.

Pie/Donut

Specifies the pie properties.



Donut Hole

Specifies the percentage the hole's thickness will take from the total radius of the donut pie circle. Available to donut pie chart only.

Pie / Donut

- **Show Pie / Donut Name**
  Specifies whether or not to show the pie name.

Static Data Label

Specifies properties of the static data labels.

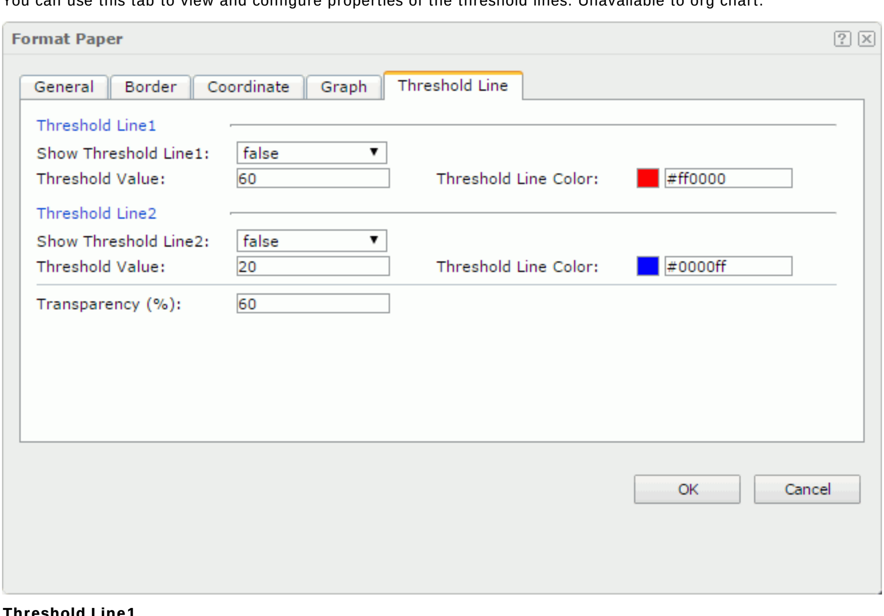
- **Show Static Data Label**
  Specifies whether or not to show the static data labels on the pies. Only when it is checked can the following static data label related properties take effect.
- **Type**
  Specifies in which way the static data labels will be displayed.
  - **Value** - Shows the value for each pie section.
  - **Category Name** - Shows the category name for each pie section.
  - **Percent** - Shows the percentage of each pie section to the total.
  - **Value Percent** - Shows the value and the percentage for each pie section.
- **Position**
  Specifies the position of the static data labels on the pies.
  - **Autofit**
    If selected, the static data labels will be displayed automatically.
  - **Sticker**
    If selected, the static data labels will be displayed beside each section of the pies.
  - **Slim Leg**
    If selected, the static data labels will be displayed beside each section and pointed by thin lines.
  - **Best Fit**
    If selected, the static data labels will be displayed at the best fit position automatically.
  - **On Slices**
    If selected, the static data labels will be displayed on each section (slice) of the pies.
- **Auto Arrange**
  Specifies whether to display the static data labels at the best position. Not supported on pie charts.

Notes:

- If the chart is a combo chart composed by areas, bars and lines, the Area, Bar, and Line tabs will be shown as sub tabs in the Graph tab.
- There is no Graph tab for indicator, gauge, bubble and bullet charts.

Threshold Line tab

You can use this tab to view and configure properties of the threshold lines. Unavailable to org chart.



Threshold Line1

Specifies the properties of the first threshold line.

- **Show Threshold Line1**
  Specifies whether or not to show the first threshold line.
- **Threshold Value**
  Specifies the value of the first threshold line.
- **Threshold Line Color**
  Specifies the color of the first threshold line.

Threshold Line2

Specifies the properties of the second threshold line.

- **Show Threshold Line2**
  Specifies whether or not to show the second threshold line.
- **Threshold Value**
  Specifies the value of the second threshold line.
- **Threshold Line Color**
  Specifies the color of the second threshold line.

Transparency

Specifies the transparency of the threshold lines.

Format Platform dialog

This dialog appears when you right-click a chart and select Format Platform from the shortcut menu. It you to format the platform of the chart and contains the following tabs:

- General tab
- Border tab
- Data tab

OK

Applies the settings and closes this dialog.

Cancel

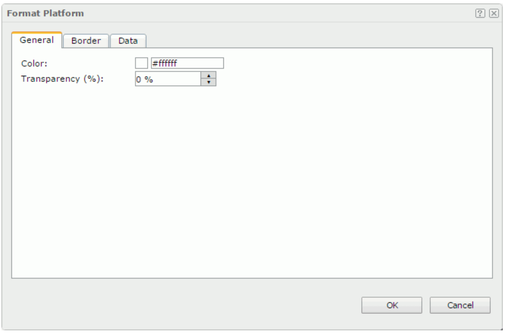
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

Specifies the color schema to fill the platform.



Color

Specifies the color with which to fill the platform.

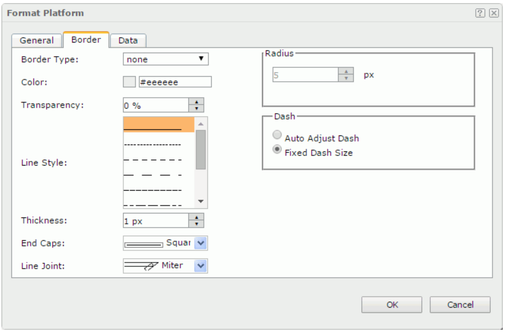
To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB, or click **More Fill Effects** to specify a gradient or an image as the fill effect in the Fill Effects dialog. If you want to make the background transparent, input Transparent in the text box.

Transparency

Specifies the transparency of the color.

Border tab

Specifies the properties for borders of the chart platform.



Border Type

Specifies the type for border of the platform.

- **None**
  The object has no visible border lines.
- **Raised**
  The object has 3D borders that appear as if they are raised off the page.
- **Recess**
  The object has 3D borders that appear as if they are pressed into the page.
- **Shadow**
  The object has two shadowed borders, beneath and to the right of the object.
- **Solid**
  The object has single-line borders (default value).

Color

Specifies the color for border of the platform.

Transparency

Specifies the transparency for color of the border.

Line Style

Specifies the line style to apply to the border of the platform.

Thickness

Specifies the thickness of the border.

End Caps

Specifies the ending style of the border line.

- **Butt**
  Ends unclosed subpaths and dash segments with no added decoration.
- **Round**
  Ends unclosed subpaths and dash segments with a round decoration that has a radius equal to half of the width of the pen.
- **Square**
  Ends unclosed subpaths and dash segments with a square projection that extends beyond the end of the segment to a distance equal to half of the line width.

Line Joint

Specifies the line joint style for the border line.

- **Miter**
  Joins path segments by extending their outside edges until they meet.
- **Round**
  Joins path segments by rounding off the corner at a radius of half the line width.
- **Bevel**
  Joins path segments by connecting the outer corners of their wide outlines with a straight segment.
- **Joint Round**
  Joins path segments by rounding off the corner at a specified radius.

Radius

Specifies the radius for the border joint of the platform border line. Available only when Line Joint is set to Joint Round.

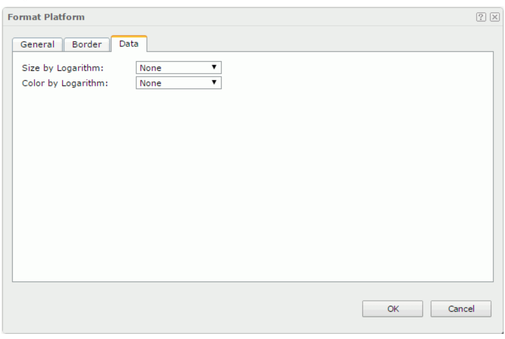
Dash

Specifies the dash size of border line.

- **Auto Adjusted Dash**
  If selected, the dash size will be adjusted automatically.
- **Fixed Dash Size**
  If selected, the dash size will be fixed size.

Data tab

Available to heat maps only. In this tab, you can specify whether to calculate the size-by/color-by summary based on logarithm functions.



Size by Logarithm

Specifies whether to calculate the size-by summary based on logarithm function.

- **None**
  Returns the real value of the field instead of any logarithmic value.
- **log x**
  Gets the logarithm of the field with base 10.
- **ln x**
  Gets the logarithm of the field with base e.

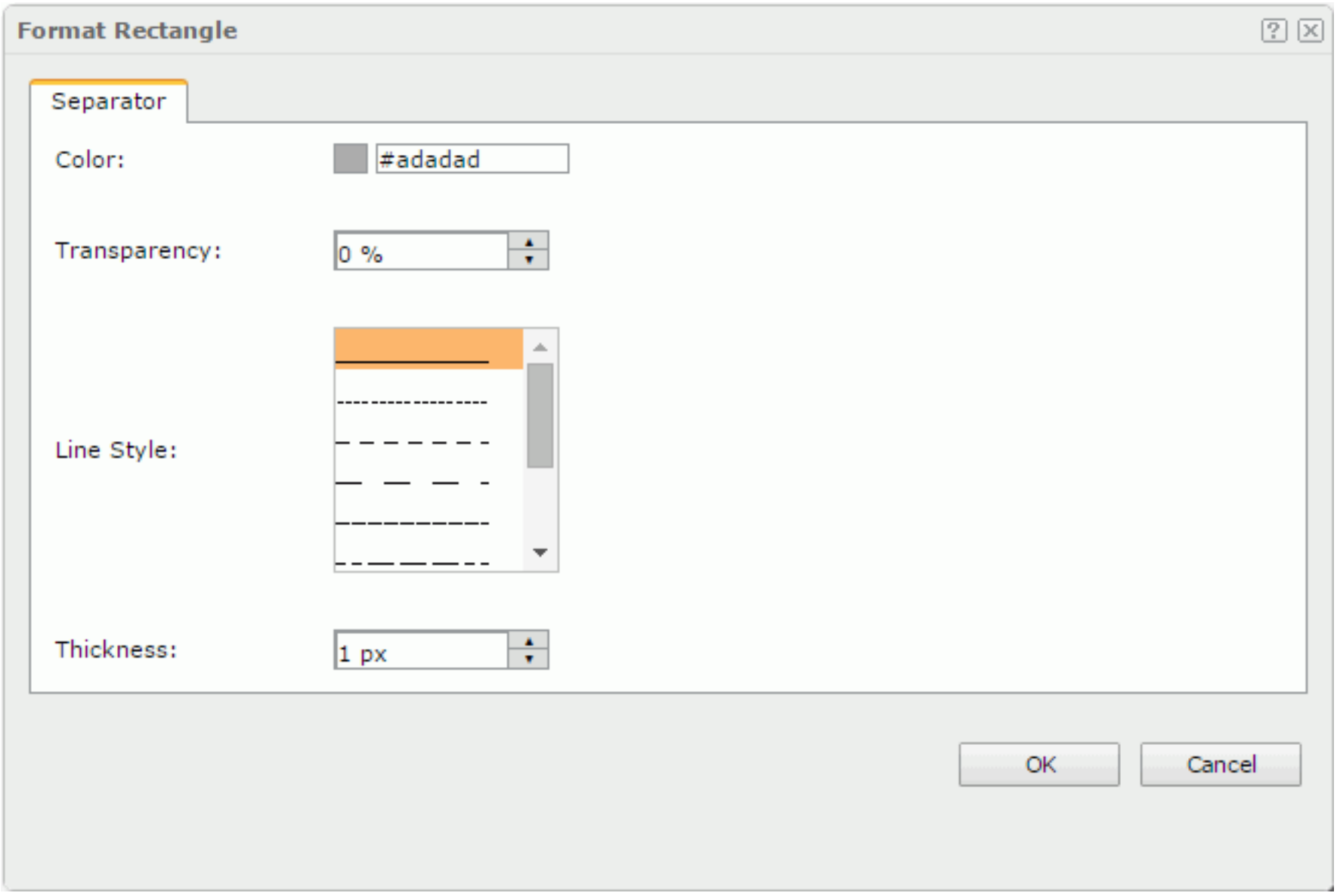
Color by Logarithm

Specifies whether to calculate the color-by summary based on logarithm function.

- **None**
  Returns the real value of the field instead of any logarithmic value.
- **log x**
  Gets the logarithm of the field with base 10.
- **ln x**
  Gets the logarithm of the field with base e.

Format Rectangle dialog

The dialog appears when you right-click on a heat map, select a group field from the Format Rectangle submenu. It enables you to customize the line properties of the rectangle separator of the specified group.



Color

Specifies the color of the separator.

Transparency

Specifies the transparency for color of the separator.

Line Style

Specifies the line style of the separator.

Thickness

Specifies the thickness of the separator, in pixels.

OK

Applies the changes and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Format Rectangle Title dialog

The dialog appears when you right-click on a heat map, select a group field from the Format Rectangle Title submenu. It helps you to format the rectangle title of the specified group, and consists of the following tabs:

- General tab
- Font tab
- Border tab

OK

Applies the changes and closes the dialog.

Cancel

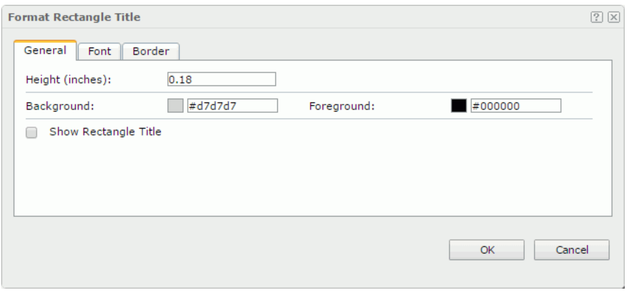
Does not retain any changes and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

Specifies the height and color properties of the rectangle title.



Height

Specifies the height of the title.

Background

Specifies the background color of the title.

To change the color, click the color indicator to select a color, or click More Colors in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Foreground

Specifies the foreground color of the title.

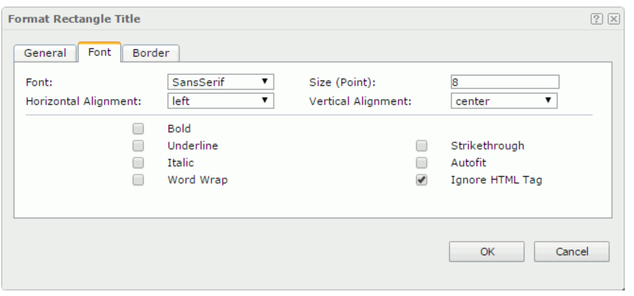
To change the color, click the color indicator to select a color, or click More Colors in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB.

Show Rectangle Title

Specifies whether to show the rectangle title.

Font tab

Specifies the font properties of the rectangle title.



Font

Specifies the font face of the text in the title.

Size

Specifies the font size of the text in the title.

Horizontal Alignment

Specifies the horizontal alignment mode of the text in the title.

Vertical Alignment

Specifies the vertical alignment mode of the text in the title.

Bold

Specifies whether to make the text bold or not.

Underline

Specifies whether the text will be underlined or not.

Italic

Specifies whether to make the text italic or not.

Word Wrap

Specifies whether or not to wrap the text to the title width.

Strikethrough

Specifies whether or not to attach a strikeout line to the text.

Autofit

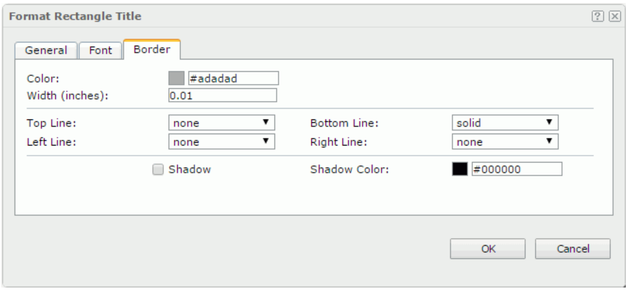
Specifies whether or not to automatically adjust the width of the title.

Ignore HTML Tag

If this option is unchecked, JReport will parse HTML tag elements in the field value while the report is to be saved as an HTML file; or the field value will appear in the HTML file the same as that in Web Report Studio (HTML tag elements in the field value, if any, will not be parsed).

Border tab

Specifies the border properties of the rectangle title.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Shadow

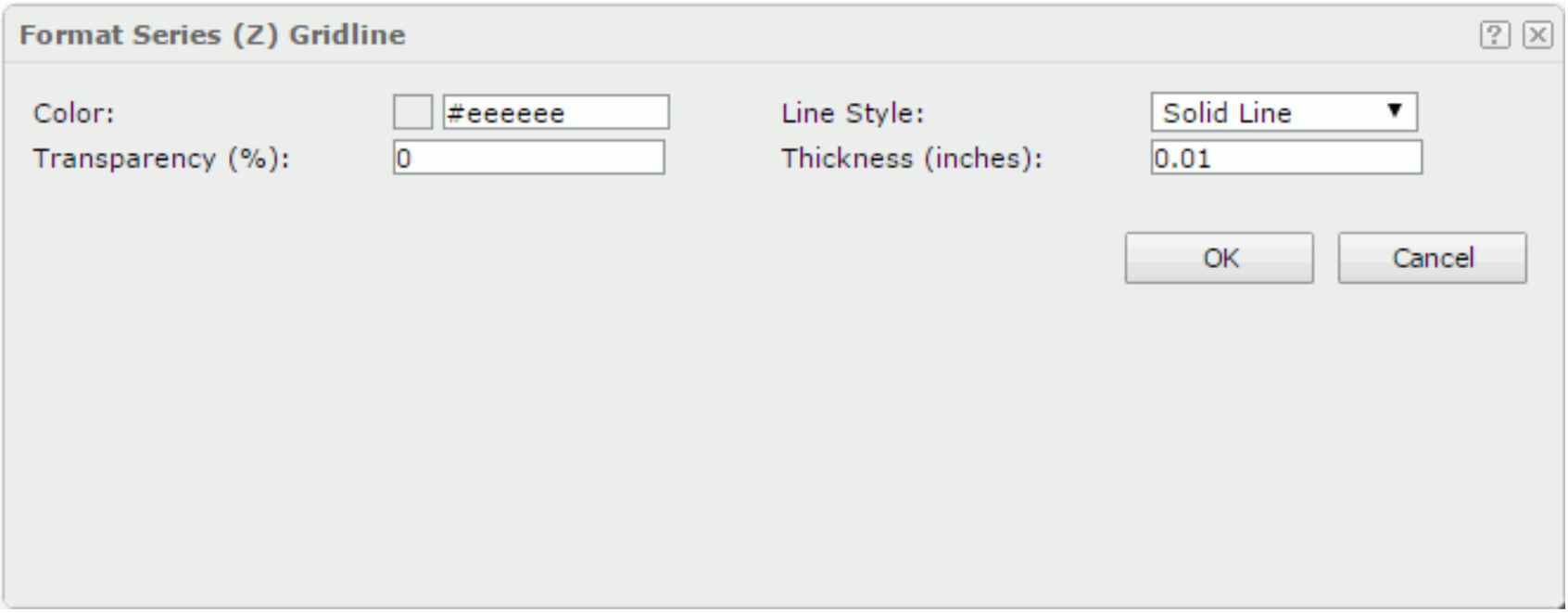
Specifies whether the borders will have a shadow effect or not.

Shadow Color

Specifies the color of the border shadow.

Format Series (Z) Gridline dialog

This dialog appears when you right-click a chart and select Format Gridlines > Format Series (Z) Gridline (available to gauge bubble chart only) from the shortcut menu. It helps you to format the gridlines shown on the series (Z) axis.



Color

Specifies the color schema of the gridlines.

Transparency

Specifies the transparency of the color that is applied to the gridlines.

Line Style

Specifies the line style of value gridlines.

Thickness

Specifies the thickness of value gridlines.

OK

Applies the changes and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Format Value (Y) Axis dialog

This dialog appears when you right-click a chart and select Format Axes (unavailable to pie, indicator, heat map and org charts) > Format Value (Y) Axis from the shortcut menu. It helps you to format the value (Y) axis of the chart and contains the following tabs:

- Axis tab
- Tick Mark tab
- Font tab
- Orientation tab
- Format tab

OK

Applies the changes and closes the dialog.

Cancel

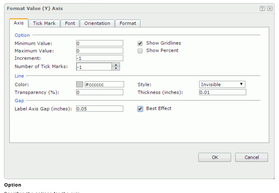
Does not retain any changes and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Axis tab

Specifies the general properties for the axis.



Option

Specifies the options for the axis.

- **Minimum Value**
  Specifies the minimum value that is to be displayed on the axis.
- **Maximum Value**
  Specifies the maximum value that is to be displayed on the axis.
- **Increment**
  Specifies the difference between two adjacent values on the axis.
- **Number of Tick Marks**
  Specifies how many tick marks to be displayed on the axis.
- **Show Gridlines**
  Specifies whether to show the horizontal gridlines in the chart.
- **Show Percent**
  Specifies whether to show the value labels on the axis in percent. Only applies to bullet, bar/bench, line, and area chart that are not 100% stacked type.

Line

Specifies the line style for the axis.

- **Color**
  Specifies the color of the axis.
- **Style**
  Specifies the style of the axis.
- **Transparency**
  Specifies the transparency for the color of the axis.
- **Thickness**
  Specifies the thickness of the axis, in pixels.

Gap

Specifies the gap properties for the labels on the axis.

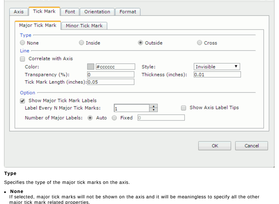
- **Label Axis Gap**
  Specifies the distance between the label and the axis, in pixels.
- **Best Effect**
  Specifies whether to adjust the labels on the axis automatically to make them placed best.

Tick Mark tab

The tab consists of two sub tabs: major tick mark tab, and minor tick mark tab.

Major Tick Mark tab

Specifies properties of the major tick marks on the axis.



Type

Specifies the type of the major tick marks on the axis.

- **None**
  If selected, major tick marks will not be shown on the axis and it will be meaningless to specify all the other major tick mark related properties.
- **Inside**
  If selected, major tick marks will be inside the chart.
- **Outside**
  If selected, major tick marks will be outside the chart.
- **Cross**
  If selected, major tick marks will be across the axis.

Line

Specifies the line properties of the major tick marks on the axis.

- **Correlate with Axis**
  If checked, the line properties of the major tick marks will correlate with that of the axis automatically.
  - **Color**
    Specifies the color of the major tick marks.
  - **Transparency**
    Specifies the transparency for the color of the major tick marks.
  - **Style**
    Specifies the type of the major tick marks.
  - **Thickness**
    Specifies the thickness of the major tick marks, in pixels.
- **Tick Mark Length**
  Specifies the length for the major tick marks, in pixels.

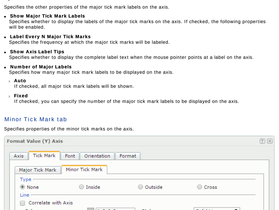
Option

Specifies the other properties of the major tick mark labels on the axis.

- **Show Major Tick Mark Labels**
  Specifies whether to display the labels of the major tick marks on the axis. If checked, the following properties will be enabled.
- **Label Every N Major Tick Marks**
  Specifies the frequency at which the major tick marks will be labeled.
- **Show Axis Label Tips**
  Specifies whether to display the complete label text when the mouse pointer points at a label on the axis.
- **Number of Major Labels**
  Specifies how many major tick mark labels to be displayed on the axis.
  - **Auto**
    If checked, all major tick mark labels will be shown.
  - **Fixed**
    If checked, you can specify the number of the major tick mark labels to be displayed on the axis.

Minor Tick Mark tab

Specifies properties of the minor tick marks on the axis.



Type

Specifies the type of the minor tick marks on the axis.

- **None**
  If selected, minor tick marks will not be shown on the axis and it will be meaningless to specify all the other minor tick mark related properties.
- **Inside**
  If selected, minor tick marks will be inside the chart.
- **Outside**
  If selected, minor tick marks will be outside the chart.
- **Cross**
  If selected, minor tick marks will be across the axis.

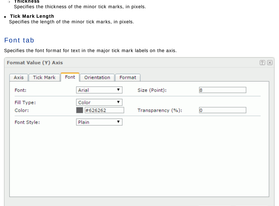
Line

Specifies the line properties of the minor tick marks on the axis.

- **Correlate with Axis**
  If checked, the line properties of the minor tick marks will correlate with that of the axis automatically.
  - **Color**
    Specifies the color of the minor tick marks.
  - **Style**
    Specifies the type of the minor tick marks.
  - **Transparency**
    Specifies the transparency for the color of the minor tick marks.
  - **Thickness**
    Specifies the thickness of the minor tick marks, in pixels.
- **Tick Mark Length**
  Specifies the length of the minor tick marks, in pixels.

Font tab

Specifies the font format for text in the major tick mark labels on the axis.



Font

Specifies the font face of the label text.

Size

Specifies the font size of the label text.

Fill Type

Specifies the fill type of the label text.

Color

Specifies the color of the label text.

Transparency

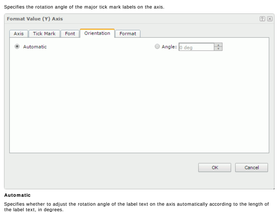
Specifies the transparency of the label text.

Font Style

Specifies the font style of the text. It can be one of the following: Plain, Bold, Italic and Bold Italic.

Orientation tab

Specifies the rotation angle of the major tick mark labels on the axis.



Automatic

Specifies whether to adjust the rotation angle of the label text on the axis automatically according to the length of the label text, in degrees.

When this option is checked by default:

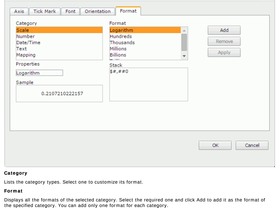
- If the text can be completely displayed horizontally, the default rotation angle will be 0.
- If the text cannot be completely displayed horizontally, the default rotation angle will be 30 anticlockwise, and the cut off part will be shown as suspension points.

Angle

Specifies to customize the rotation angle of the label text on the axis.

Format tab

Specifies the data format of the major tick mark labels on the axis.



Category

Lists the category types. Select one to customize its format.

Format

Displays all the formats of the selected category. Select the required one and click Add to add it as the format of the specified category. You can add only one format for each category.

Properties

Displays the properties of the format you select. If the formats listed in the Format box cannot meet your requirement, define the format in the text field and then click Add to add it as the format of the specified category.

Sample

Displays the selected format effects.

Stack

Lists all the formats you select from different categories.

Add

Adds a format to the Stack list box.

Remove

Removes a format from the Stack list box.

Apply

Applies the specified format in the Stack list box to the major tick mark labels.

Format Value (Y) Gridline dialog

This dialog appears when you right-click a chart and select Format Gridlines (unavailable to pie, indicator, heat map and org charts) > Format Value (Y) Gridline from the shortcut menu. It helps you to format the gridlines shown on the value (Y) axis.

Color

Specifies the color schema of the gridlines.

Transparency

Specifies the transparency of the color that is applied to the gridlines.

Line Style

Specifies the line style of value gridlines.

Thickness

Specifies the thickness of value gridlines.

OK

Applies the changes and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Format Value (Y2) Axis dialog

This dialog appears when you right-click a chart and select Format Axes (unavailable to pie, indicator, heat map and gantt charts) > Format Value (Y2) Axis from the shortcut menu. It helps you to format the value (Y2) axis of the chart and contains the following tabs:

- Axis tab
- Tick Mark tab
- Font tab
- Orientation tab
- Format tab

OK

Applies the changes and closes the dialog.

Cancel

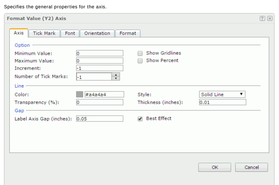
Does not retain any changes and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Axis tab

Specifies the general properties for the axis.



Option

Specifies the options for the axis.

- **Minimum Value**
  Specifies the minimum value that is to be displayed on the axis.
- **Maximum Value**
  Specifies the maximum value that is to be displayed on the axis.
- **Increment**
  Specifies the difference between two adjacent values on the axis.
- **Number of Tick Marks**
  Specifies how many tick marks to be displayed on the axis.
- **Show Gridlines**
  Specifies whether to show the horizontal gridlines in the chart.
- **Show Percent**
  Specifies whether to show the value labels on the axis in percent. Only applies to bullet, bar/bench, line, and area chart that are not 100% stacked type.

Line

Specifies the line style for the axis.

- **Color**
  Specifies the color of the axis.
- **Style**
  Specifies the style of the axis.
- **Transparency**
  Specifies the transparency for the color of the axis.
- **Thickness**
  Specifies the thickness of the axis, in pixels.

Gap

Specifies the gap properties for the labels on the axis.

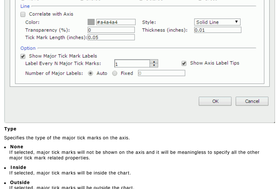
- **Label Axis Gap**
  Specifies the distance between the label and the axis, in pixels.
- **Best Effect**
  Specifies whether to adjust the labels on the axis automatically to make them placed best.

Tick Mark tab

The tab consists of two sub tabs: major tick mark tab, and minor tick mark tab.

Major Tick Mark tab

Specifies properties of the major tick marks on the axis.



Type

Specifies the type of the major tick marks on the axis.

- **None**
  If selected, major tick marks will not be shown on the axis and it will be meaningless to specify all the other major tick mark related properties.
- **Inside**
  If selected, major tick marks will be inside the chart.
- **Outside**
  If selected, major tick marks will be outside the chart.
- **Cross**
  If selected, major tick marks will be across the axis.

Line

Specifies the line properties of the major tick marks on the axis.

- **Correlate with Axis**
  If checked, the line properties of the major tick marks will correlate with that of the axis automatically.
  - **Color**
    Specifies the color of the major tick marks.
  - **Transparency**
    Specifies the transparency for the color of the major tick marks.
  - **Style**
    Specifies the type of the major tick marks.
  - **Thickness**
    Specifies the thickness of the major tick marks, in pixels.
- **Tick Mark Length**
  Specifies the length for the major tick marks, in pixels.

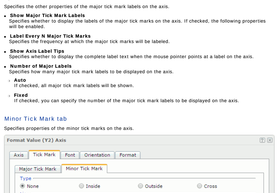
Option

Specifies the other properties of the major tick mark labels on the axis.

- **Show Major Tick Mark Labels**
  Specifies whether to display the labels of the major tick marks on the axis. If checked, the following properties will be enabled.
- **Label Every N Major Tick Marks**
  Specifies the frequency at which the major tick marks will be labeled.
- **Show Axis Label Tips**
  Specifies whether to display the complete label text when the mouse pointer points at a label on the axis.
- **Number of Major Labels**
  Specifies how many major tick mark labels to be displayed on the axis.
  - **Auto**
    If checked, all major tick mark labels will be shown.
  - **Fixed**
    If checked, you can specify the number of the major tick mark labels to be displayed on the axis.

Minor Tick Mark tab

Specifies properties of the minor tick marks on the axis.



Type

Specifies the type of the minor tick marks on the axis.

- **None**
  If selected, minor tick marks will not be shown on the axis and it will be meaningless to specify all the other minor tick mark related properties.
- **Inside**
  If selected, minor tick marks will be inside the chart.
- **Outside**
  If selected, minor tick marks will be outside the chart.
- **Cross**
  If selected, minor tick marks will be across the axis.

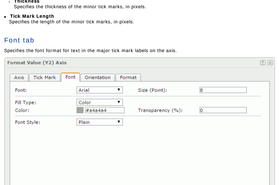
Line

Specifies the line properties of the minor tick marks on the axis.

- **Correlate with Axis**
  If checked, the line properties of the minor tick marks will correlate with that of the axis automatically.
  - **Color**
    Specifies the color of the minor tick marks.
  - **Style**
    Specifies the type of the minor tick marks.
  - **Transparency**
    Specifies the transparency for the color of the minor tick marks.
  - **Thickness**
    Specifies the thickness of the minor tick marks, in pixels.
- **Tick Mark Length**
  Specifies the length of the minor tick marks, in pixels.

Font tab

Specifies the font format for text in the major tick mark labels on the axis.



Font

Specifies the font face of the label text.

Size

Specifies the font size of the label text.

Fill Type

Specifies the fill type of the label text.

Color

Specifies the color of the label text.

Transparency

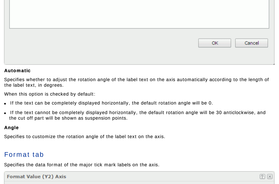
Specifies the transparency of the label text.

Font Style

Specifies the font style of the text. It can be one of the following: Plain, Bold, Italic and Bold Italic.

Orientation tab

Specifies the rotation angle of the major tick mark labels on the axis.



Automatic

Specifies whether to adjust the rotation angle of the label text on the axis automatically according to the length of the label text, in degrees.

When this option is checked by default:

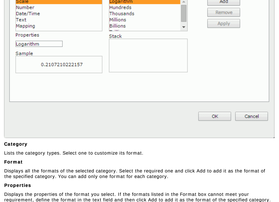
- If the text can be completely displayed horizontally, the default rotation angle will be 0.
- If the text cannot be completely displayed horizontally, the default rotation angle will be 30 anticlockwise, and the cut off part will be shown as suspension points.

Angle

Specifies to customize the rotation angle of the label text on the axis.

Format tab

Specifies the data format of the major tick mark labels on the axis.



Category

Lists the category types. Select one to customize its format.

Format

Displays all the formats of the selected category. Select the required one and click Add to add it as the format of the specified category. You can add only one format for each category.

Properties

Displays the properties of the format you select. If the formats listed in the Format box cannot meet your requirement, define the format in the text field and then click Add to add it as the format of the specified category.

Sample

Displays the selected format effects.

Stack

Lists all the formats you select from different categories.

Add

Adds a format to the Stack list box.

Remove

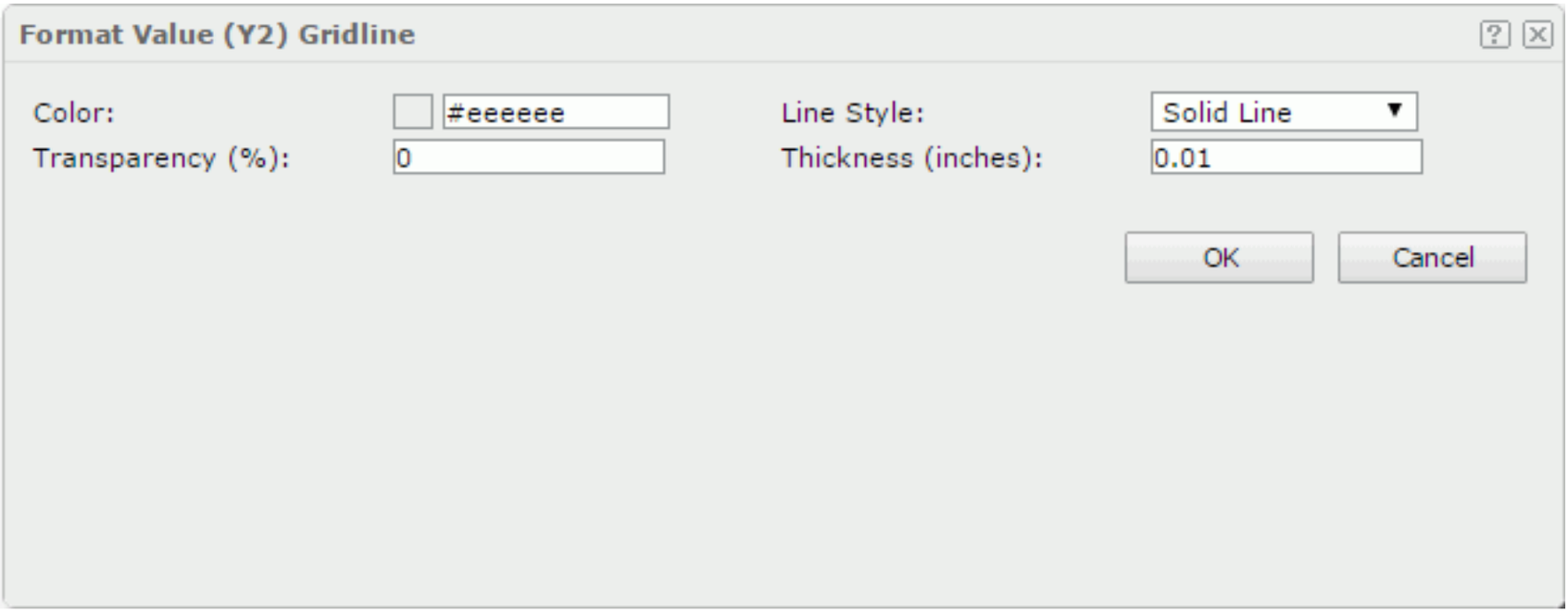
Removes a format from the Stack list box.

Apply

Applies the specified format in the Stack list box to the major tick mark labels.

Format Value (Y2) Gridline dialog

This dialog appears when you right-click a chart and select Format Gridlines (unavailable to pie, indicator, heat map and org charts) > Format Value (Y2) Gridline from the shortcut menu. It helps you to format the gridlines shown on the value (Y2) axis.



Color

Specifies the color schema of the gridlines.

Transparency

Specifies the transparency of the color that is applied to the gridlines.

Line Style

Specifies the line style of value gridlines.

Thickness

Specifies the thickness of value gridlines.

OK

Applies the changes and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Format Wall dialog

This dialog appears when you right-click a chart and select Format Walls > Format Wall from the shortcut menu. It helps you to format the wall of the chart and contains the following tabs:

- General tab
- Border tab

OK

Applies the changes and closes the dialog.

Cancel

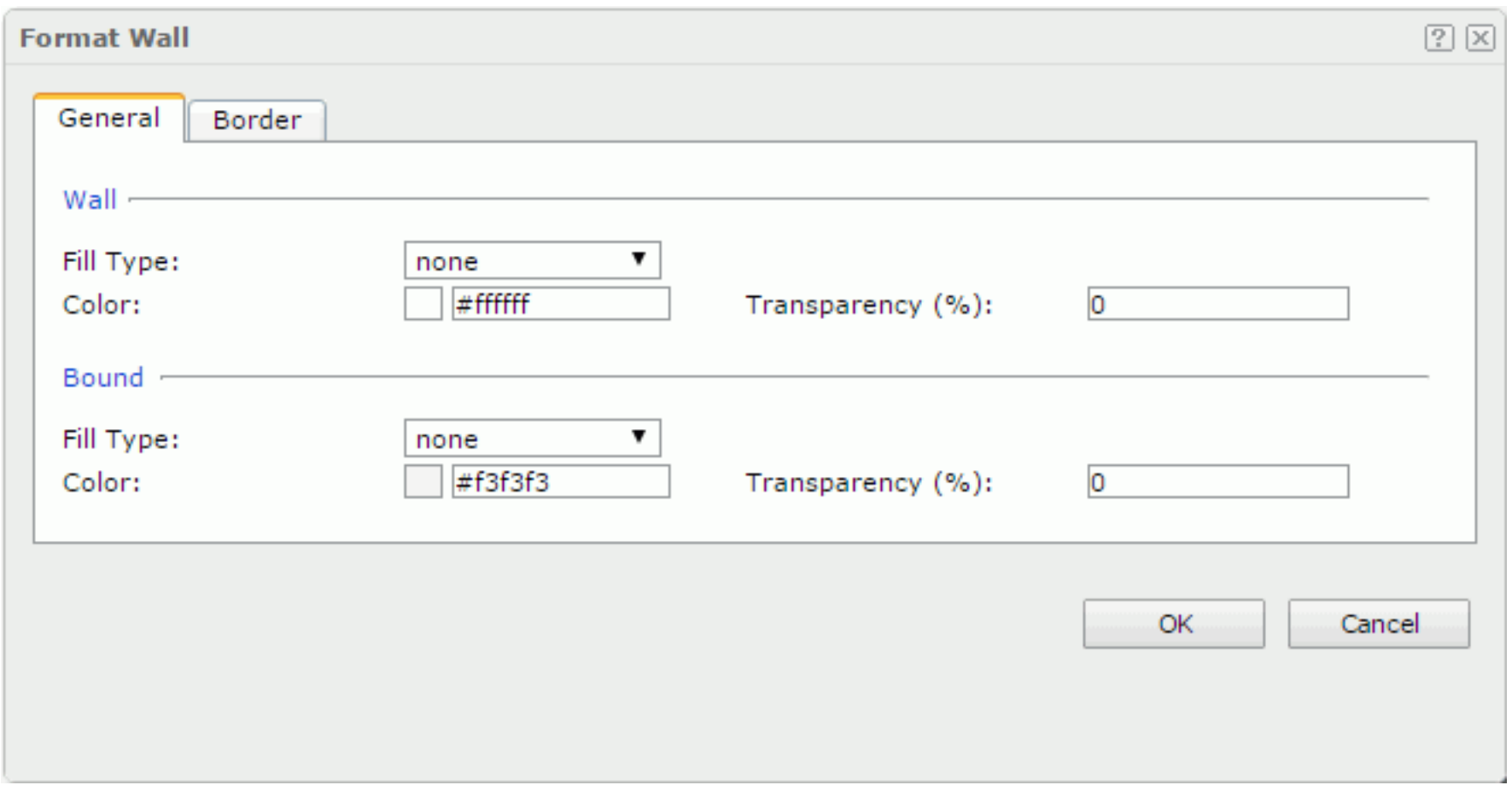
Does not retain any changes and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the wall.



Wall

Specifies the color or the fill effect of the wall of the chart.

- **Fill Type**
  Specifies the fill type of the wall.
- **Color**
  Specifies the color of the wall.
- **Transparency**
  Specifies the transparency of the color schema that is applied to the wall of the chart.

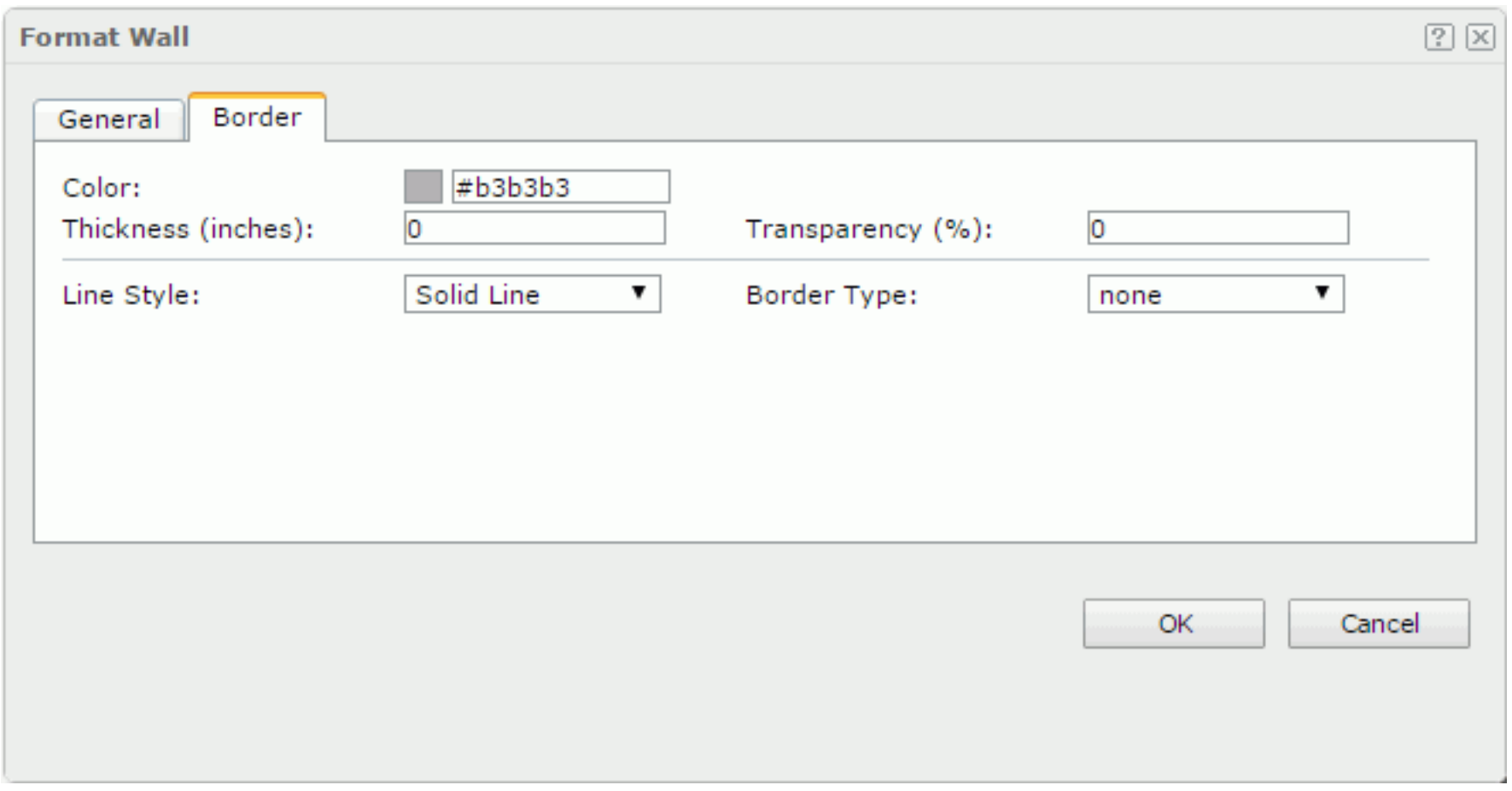
Bound

Specifies the fill color or effect to compound with the basic fill which is set in the Wall area.

- **Fill Type**
  Specifies the fill type of the wall.
- **Color**
  Specifies the fill color schema or effect to compound with the basic fill which is set for the walls of a chart in the Wall area.
- **Transparency**
  Specifies the transparency of the color schema that is applied here.

Border tab

This tab shows information about borders of the wall.



Color

Specifies the color schema for the border of the wall.

Transparency

Specifies the transparency of the color schema that is applied to the border of the wall.

Thickness

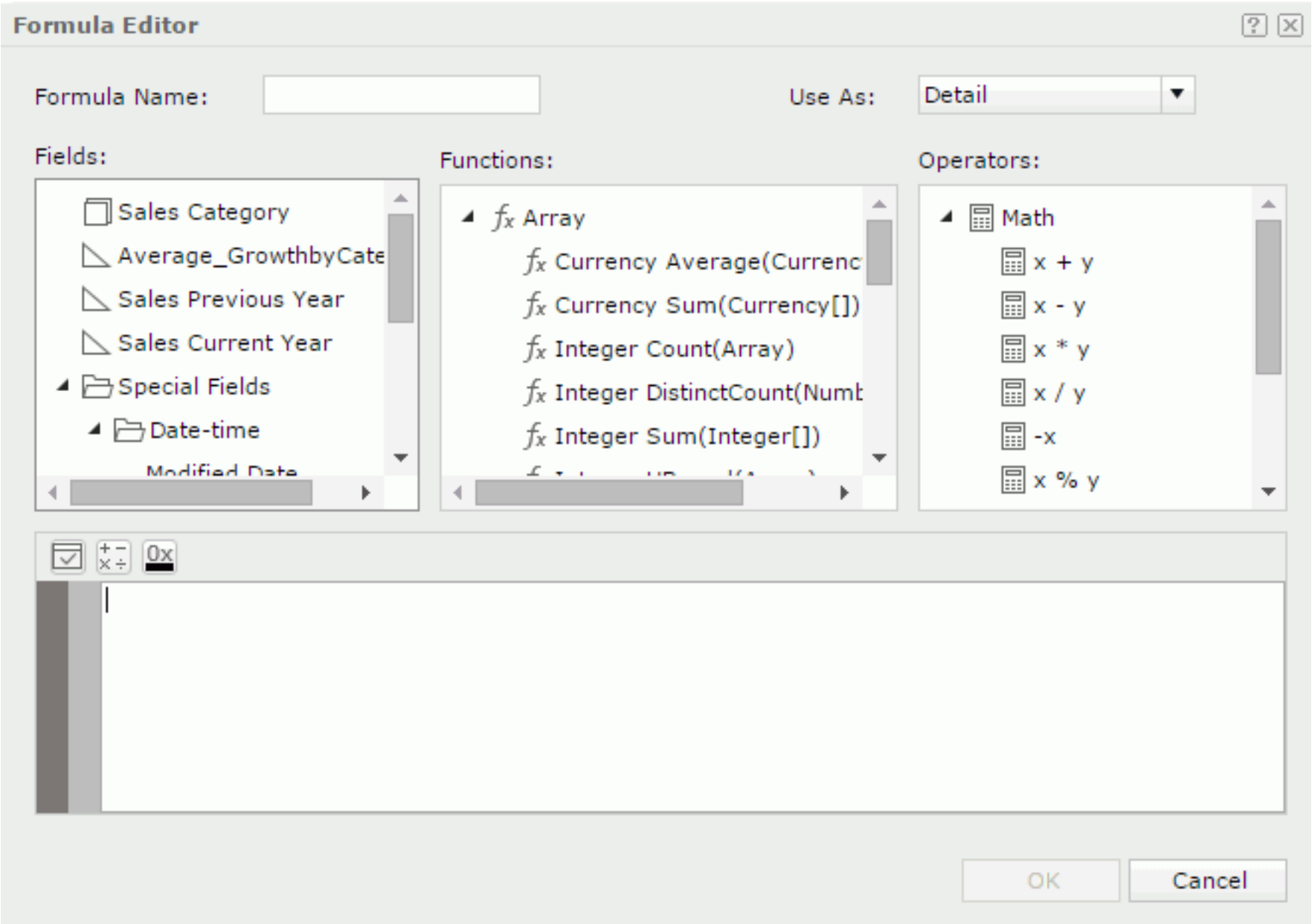
Specifies the thickness for the border of the wall.

Line Style

Specifies the line style for the border of the wall.

Formula Editor dialog

This dialog appears when you expand the Dynamic Resource > Formulas node in the Resources panel of Web Report Studio or the Resources box of the Web Report Wizard, then click <Add Formula•gt;, right-click a dynamic formula and select Edit from the shortcut menu, or click fx in the value cell and select <Edit Expression> or <Create Formula> from the value drop-down list when setting property values in the Inspector panel. It helps you to create or edit a dynamic formula in a web report, or create an expression to control the value of the specified property.



Formula Name

Specifies the name of the formula. Disabled when creating an expression.

Use As

Specifies to use the formula as one of the following types: Group, Detail, or Aggregation. Disabled when creating an expression.

Which type a dynamic formula can be used as depends on the following:

- Any formula can be used as Detail.
- Record level pass 1 formula can be used as Group.
- Comparable and record level pass 1 formula can be used as Group.
- Group level formula can be used as Aggregation.

Fields box

Displays a list of those fields that are available to formulas. The fields include the group and detail objects in the current business view, and the dynamic formulas that have been created in the report. You can select one field and double-click it to insert the field into the editing box at the insertion point.

Functions box

Displays a list of JReport functions that are available to formulas. When you select one function and double-click it, JReport will insert the selected function into the editing box at the insertion point completely with its required syntax items (parentheses, commas, and so on).

For details about usage of the functions, refer to Built-in functions.

Operators box

Displays a list of operators that are available to formulas. Select one operator and double-click it to insert the selected operator into the editing box at the insertion point.

For details about usage of the operators, refer to Operators.

Editing box

In this box, you can build and edit your formula. There are several ways to work with formulas:

- Select formula components from the Fields, Functions and Operators boxes in the Formula Editor, and then double-click the components, JReport will then insert them in the formula;
- Type your formula in the editing box directly;
- Use the above two methods together;
- Paste formula text from the text document of other programs.

Note: If you refer to any field in the formula, the reference name for that field will be prefixed with an @ sign. If the field name contains spaces, the reference name in formula will be quoted with double-quotation marks (""). For example, if the field name is Customer Name, then the reference name will be @"Customer Name".

Tests the syntax of your formula. If the syntax is incorrect, JReport provides an opportunity to correct the errors.

Selects a general operator to be used in the editing box.

A color palette is provided for you to insert the HEX code of a color simply by clicking the corresponding color in the color palette instead of inputting the HEX code manually.

OK

Creates or edits the formula and closes the dialog.

Cancel

Cancels the creation or edition of the formula and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Related topics:

- Using dynamic resources

Group Footer Properties dialog

The dialog appears when you click anywhere in a table group footer row, then click the button that appears on the group footer to select the row, right-click on the row and select Properties from the shortcut menu. It helps you to specify the properties of the group footer and contains the following tabs:

- General tab
- Border tab

OK

Applies the settings for the group footer properties and closes this dialog.

Cancel

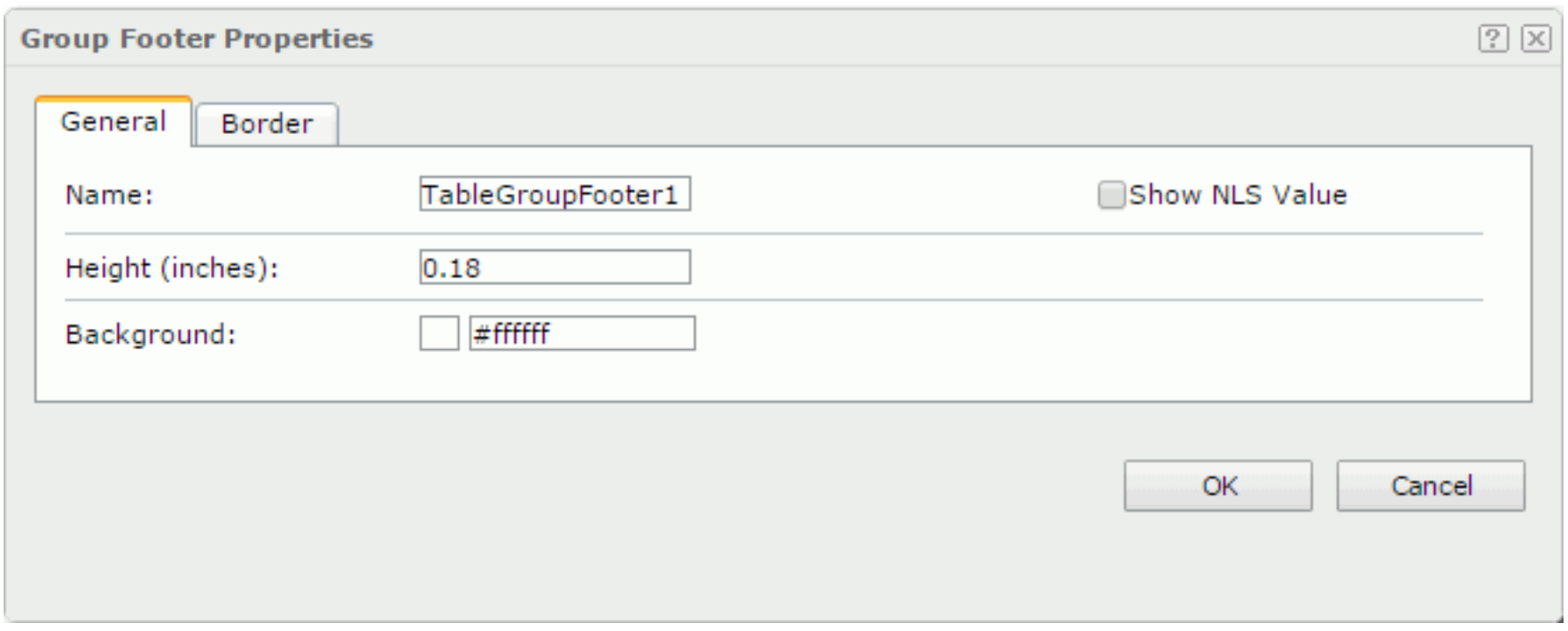
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the group footer.



Name

Specifies the display name of the group footer.

Show NLS Value

Specifies to show the translated name of the display name of the group footer in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the group footer is not modified.

Height

Specifies the height of the group footer.

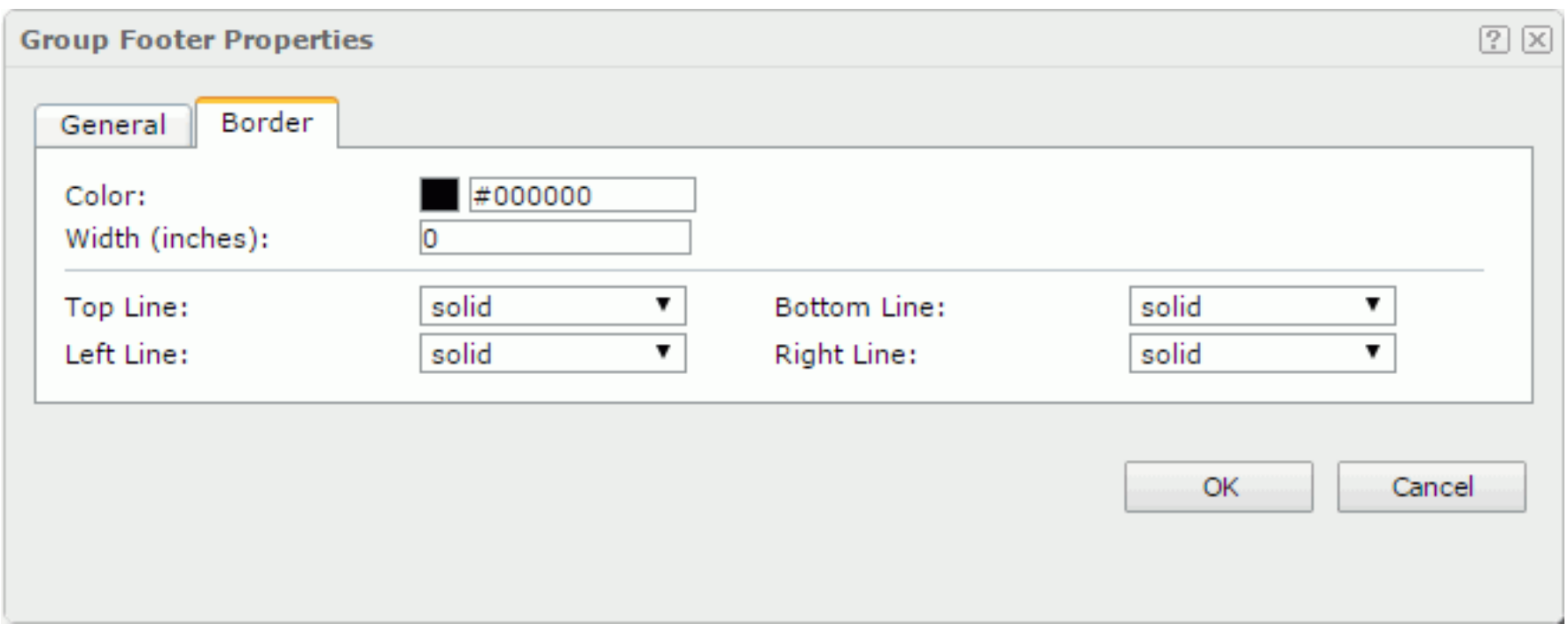
Background

Specifies the background color of the group footer.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB, or click **More Fill Effects** to specify a gradient or an image as the fill effect in the Fill Effects dialog. If you want to make the background transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the group footer.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Group Header Properties dialog

The dialog appears when you click anywhere in a table group header row, then click the button that appears on the group header to select the row, right-click on the row and select Properties from the shortcut menu. It helps you to specify the properties of the group header and contains the following tabs:

- General tab
- Border tab

OK

Applies the settings for the group header properties and closes this dialog.

Cancel

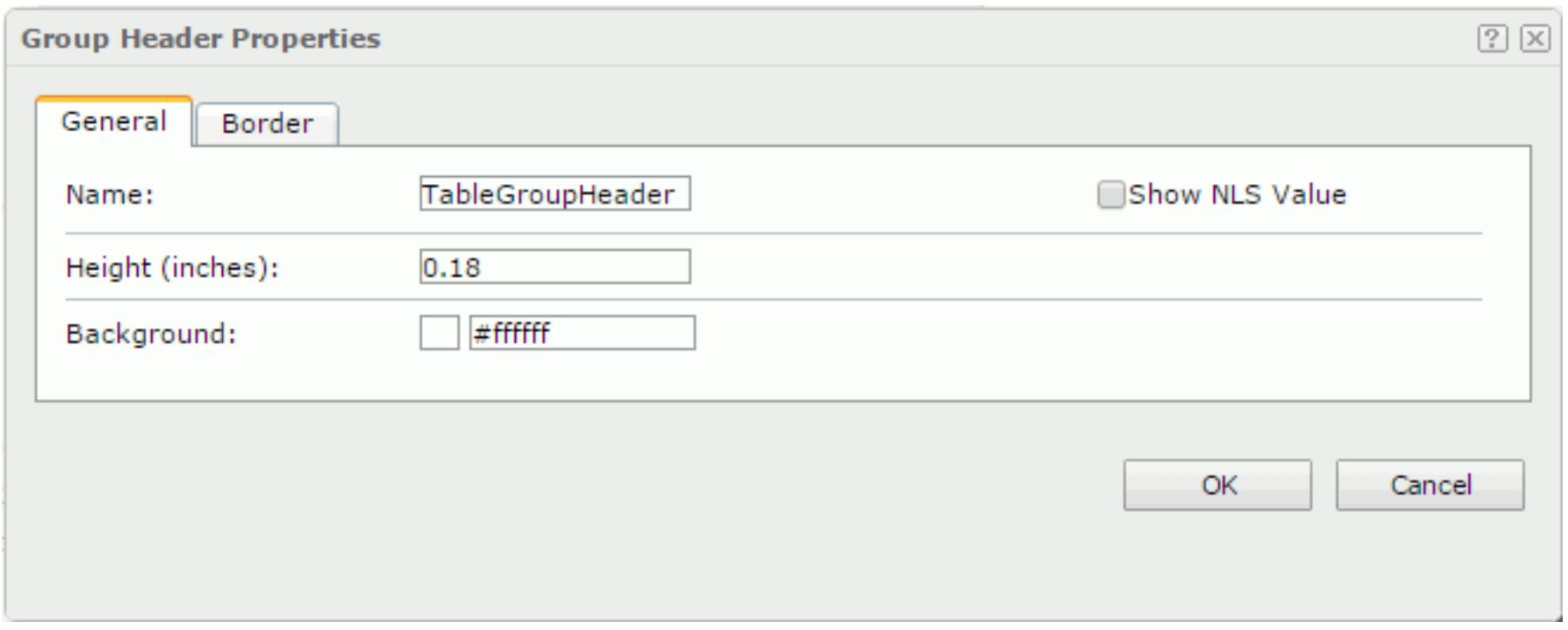
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the group header.



Name

Specifies the display name of the group header.

Show NLS Value

Specifies to show the translated name of the display name of the group header in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the group header is not modified.

Height

Specifies the height of the group header.

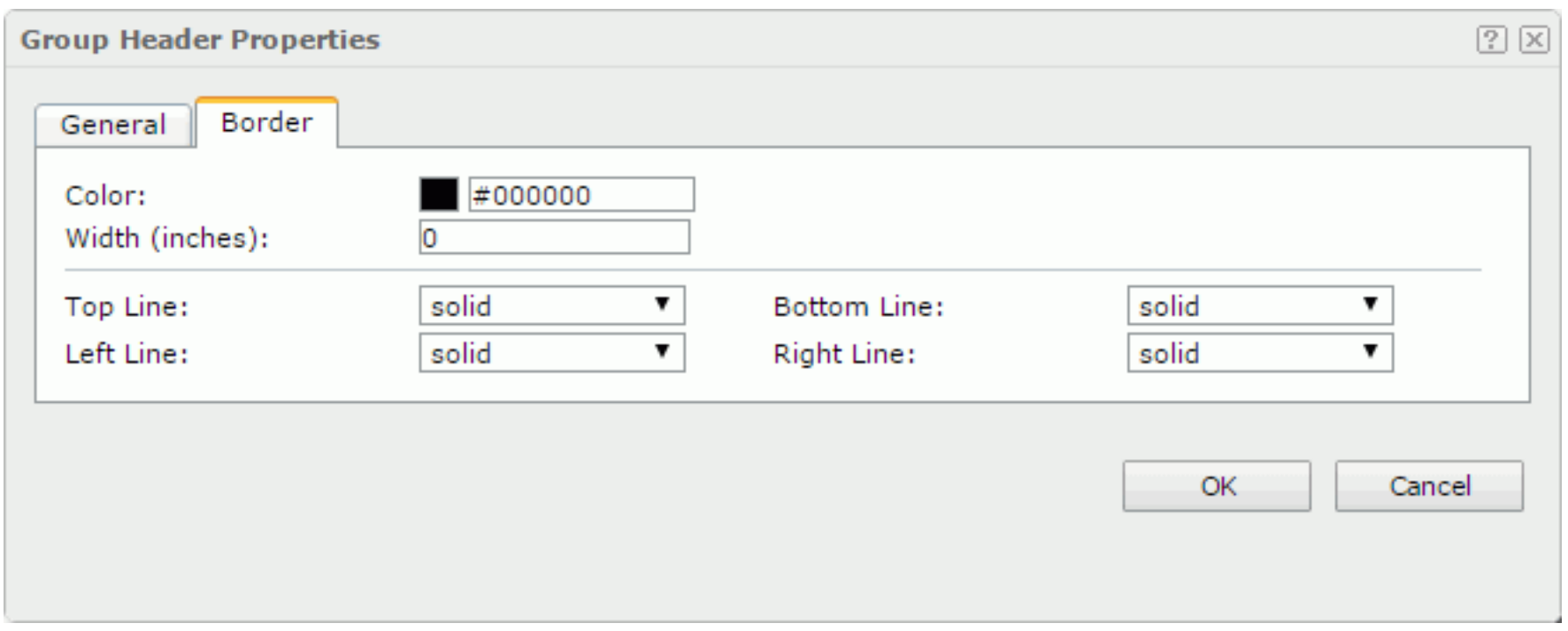
Background

Specifies the background color of the group header.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB, or click **More Fill Effects** to specify a gradient or an image as the fill effect in the Fill Effects dialog. If you want to make the background transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the group header.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

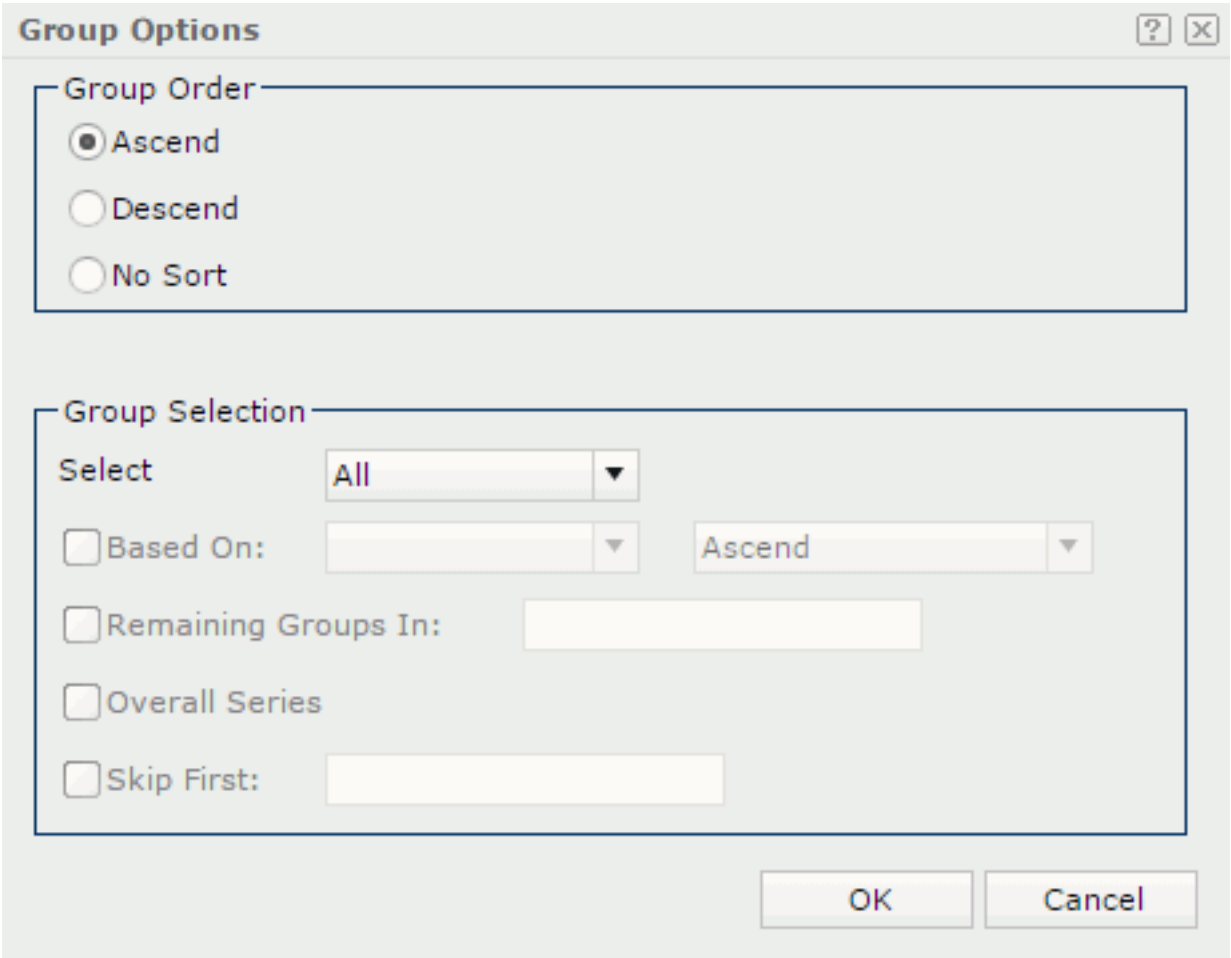
Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Group Options dialog

This dialog appears after you click the Top N button above the Group box in the Bind Data screen of chart in the Web Report Wizard, Insert Chart dialog, Chart Wizard, or To Chart dialog when the chart type is heat map. It helps you to set the sort order of the group values and define the number of the group values that will be displayed in the chart.



Group Order

Specifies in which manner to sort the group values.

- **Ascend**
  Values will be sorted in an ascending order (A, B, C).
- **Descend**
  Values will be sorted in a descending order (C, B, A).
- **No Sort**
  Values will be sorted in the original order in database.

Group Selection

Specifies the number of group values that will be displayed in the chart.

- **Select**
  Specifies the Select N condition to define the number of the group values that will be displayed.
  - **All**
    If selected, all group values in the chart will be displayed.
  - **Top N**
    If selected, specify a number in the field to the right and the first N group values in the chart will be displayed. You can also select a parameter from the drop-down list, which returns an integer, and then specify a value for the parameter when viewing the report to dynamically define the Top N condition.
  - **Bottom N**
    If selected, specify a number in the field to the right and the last N group values in the chart will be displayed. You can also select a parameter from the drop-down list which returns an integer, and then specify a value for the parameter when viewing the report to dynamically define the Bottom N condition.
- **Based On**
  If checked, the group values will be sorted by values of the summary selected with the direction specified. If unchecked, the group values will be sorted by the order specified in the Group Order box of the dialog.
- **Remaining Groups In**
  Enabled when Top N or Bottom N is selected from the Select drop-down list. Check this option and then type a character string in the text field to group all the group values beyond the top/bottom N range.
- **Overall Series**
  Not supported on group values.
- **Skip First**
  If you check the Skip First option and input a number M in the text field to the right, then the first M group values in the chart will be skipped and the Select N condition will take effect beginning with the M+1 value. The skipped values will be included in the remaining group together with all the group values beyond the top/bottom N range.

OK

Accepts the changes and exits the dialog.

Cancel

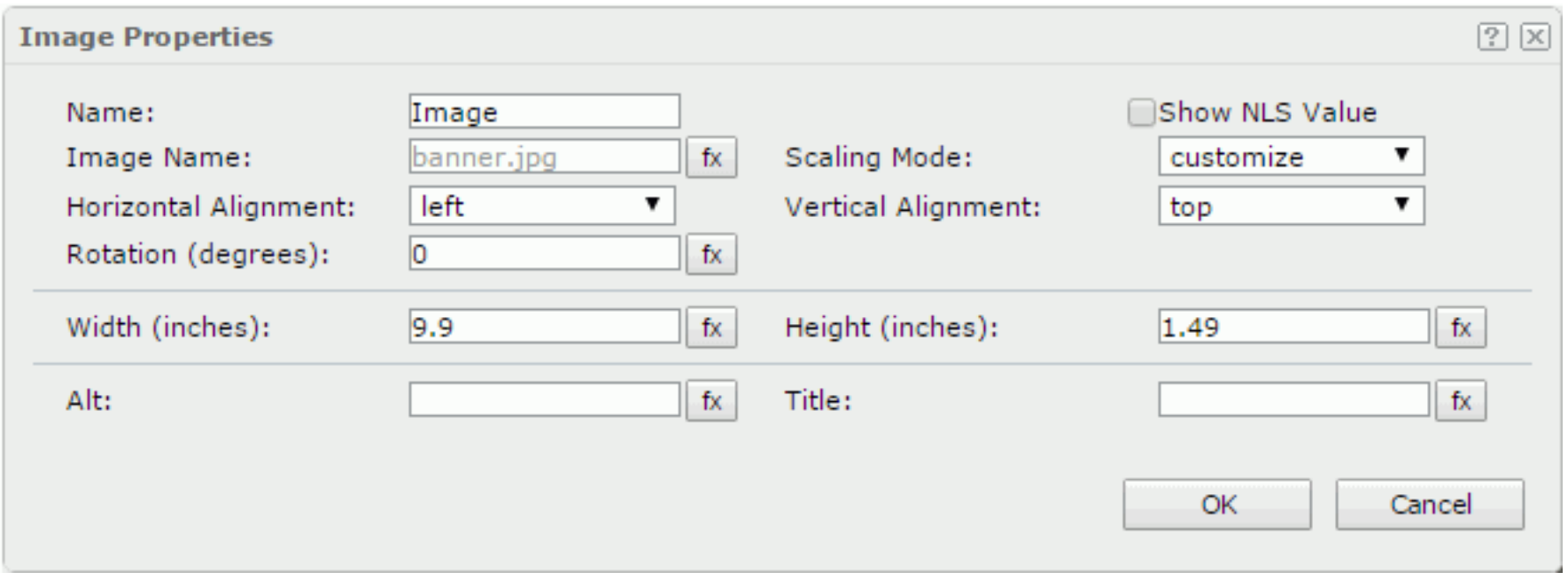
Discards the changes and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Image Properties dialog

The dialog appears when you right-click an image and select Properties from the shortcut menu. It helps you to edit the properties of the image.



Name

Specifies the display name of the image, which will be shown on the shortcut menu of the image.

Show NLS Value

Specifies to show the translated name of the display name of the image in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the image is not modified.

Image Name

Specifies the image file name.

fx

Indicates the value of the option can be controlled by a formula. Click this button and select a formula from the drop-down list, or click <Edit Expression> to create an expression using the Formula Editor.

Scaling Mode

Specifies the scaling mode for the image. The mode can be:

- **actual size**: The image will be shown in its actual size.
- **customize**: The image size will be equal to the image field size.
- **fit image**: The image will be scaled largest to wholly show in the image field.
- **fit width**: The image will be scaled largest to fit the width of the image field.
- **fit height**: The image will be scaled largest to fit the height of the image field.

Horizontal Alignment

Specifies the horizontal alignment of the image in its container.

Vertical Alignment

Specifies the vertical alignment of the image in its container.

Rotation

Rotates the image at a specified angle in degrees. The following is the meaning of different values:

- 0 - No rotation.
- Positive value - Rotate the image clockwise.
- Negative value - Rotate the image anticlockwise.

Note: When you rotate an image, the rectangle that holds the image maintains its original size, which may result in that the image exceeds the field border and therefore the parts that extend outside of the border will be cut off.

Width

Specifies the width of the image.

Height

Specifies the height of the image.

Alt

Specifies the alternate text which will be shown if the image cannot be displayed.

Title

Specifies tip information about the image, which will be displayed when the mouse cursor hovers on the image.

OK

Applies the settings and closes this dialog.

Cancel

Cancels the settings and closes this dialog.

?

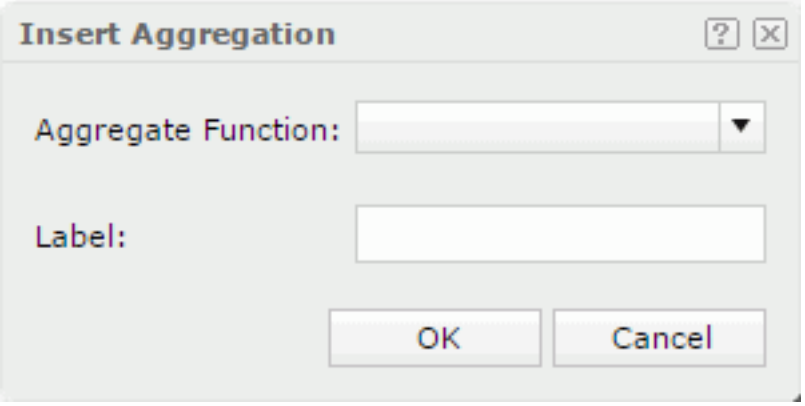
Displays the help document about this feature.

x

Ignores the setting and closes this dialog.

Insert Aggregation dialog

This dialog helps you to insert the selected field as a new aggregate field in a crosstab.



Aggregate Function

Specifies the function to summarize the selected field. For details about each function, see Math functions.

If the selected field has been predefined with an aggregate function, the option is Aggregation. The name of the field is displayed in the text box and cannot be edited.

Label

Specifies the display name of the aggregate field, which will label the newly created summaries in the crosstab. You can leave it blank.

OK

Inserts the aggregation and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Related topics:

- Inserting aggregate fields by dragging

Insert Chart dialog

This dialog appears when you click drag Chart from the Components panel to the destination. It helps you to insert a chart to a report, and varies with different chart types: common chart types, organization chart, heat map.

OK
Inserts a chart in the report and closes the dialog.

Cancel
Cancels the insertion and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

For common chart types

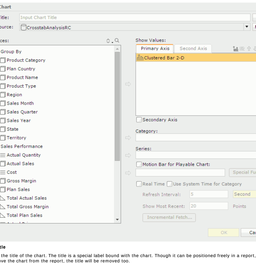


Chart Title
Specifies the title of the chart. The title is a special label bound with the chart. Though it can be positioned freely in a report, once you remove the chart from the report, the title will be removed too.

Specifies the text properties of the chart title.

- **Font**
  Lists all the available font faces that can be selected to apply to the title.
- **Font Style**
  Specifies the font style of the title. It can be one of the following: plain, bold, italic, and bold italic.
- **Size**
  Specifies the font size of the title.
- **Align**
  Specifies the position of the title to be left, right, center or justify.
- **Font Color**
  Specifies the font color of the title.
- **Background Color**
  Specifies the background color of the title.

Data Source
Specifies the business view in the current catalog on which the chart will be built.

Filter
Opens the Query Filter dialog to specify the filter you want to apply to the selected business view.

Resources
Displays the resources that can be added to the chart.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.
- **.**
  Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed-text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
- **.**
  Highlights the previous matched text.
- **.**
  Highlights the next matched text.

Adds the selected resource to be displayed in the chart.

Show Values
Lists the values you want to show in the chart. For a real time chart, the values you add must be of Numeric type and cannot be aggregation objects.

- **Primary Axis**
  Adds a chart type to the primary axis.
- **Secondary Axis**
  Adds a chart type to the secondary axis. Active only when the option Secondary Axis is checked.
- **X Axis**
  Lists the value you want to show on the X axis of the bubble chart.
- **Y Axis**
  Lists the value you want to show on the Y axis of the bubble chart.
- **Radius**
  Lists the value you want to show as the bubble radius.

Adds a combo chart to the Primary Axis or Secondary Axis.

Opens the Edit Additional Value dialog to edit the selected additional value.

Moves the selected view element one level up.

Moves the selected view element one level down.

Adds a new pair of Y Axis and Radius for the bubble chart.

Secondary Axis
Specifies whether to show the secondary axis in the chart.

Category
Lists the group object that will be displayed on the category axis of the chart.
For a real time chart, if no object is specified on the category axis, Use System Refresh Time will be automatically displayed in the Category text box, namely, the time at which the chart refreshes itself will be used as the category value.

Series
Lists the group object that will be displayed on the series axis of the chart. Not available to real time chart.

Opens the Category Options dialog or Series Options dialog to define the sort order of the category or series values and specify the number of the category or series values that will be displayed in the chart.

Motion Bar for Playable Chart
Lists the group object you want to use as the motion field. A motion field can only be of Integer, Date or Time data type. Available to single bar, bench and bubble chart types only.

- **Special Function**
  Available only when the motion field is of Date data type. Click it to define the special function.
  - **Field**
    Displays on which field the special function will be applied.
  - **Function**
    Specifies the special function to the field.
  - **OK**
    Accepts the special function settings and leaves the dialog.
  - **Cancel**
    Cancels the special function settings and leaves the dialog.

Real Time
Specifies to run the chart in real time mode, which means it will be updated automatically by using real time data. Available to single bar, bench, line, and area chart types only.

- **Use System Time for Category**
  Specifies to use the time at which the chart refreshes itself as the category value.
- **Refresh Interval**
  Specifies the time interval at which the chart will get data and refresh itself automatically.
- **Show Most Recent N Points**
  Specifies the number of recent points that will be kept for the real time data on the chart.
- **Incremental Fetch**
  Opens the Unique Key dialog to configure a unique key for the real time chart.

Removes the selected resource.

For organization chart

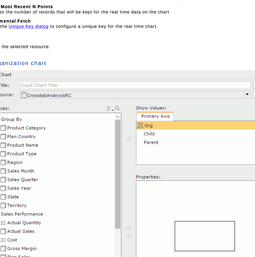


Chart Title
Specifies the title of the chart. The title is a special label bound with the chart. Though it can be positioned freely in a report, once you remove the chart from the report, the title will be removed too.

Specifies the font properties of the chart title.

Data Source
Specifies the business view in the current catalog on which the chart will be built.

Filter
Opens the Query Filter dialog to specify the filter you want to apply to the selected business view.

Resources
Displays the resources that can be added to the chart.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

Launches the quick search toolbar to search for view elements.

Show Values

- **Primary Axis**
  Select Org from the chart type drop-down menu.
- **Child**
  Adds a field from the Resources box which identifies the entity by selecting both the field and Child and then clicking .
- **Parent**
  Adds a field from the Resources box which shows the "reporting to" relationship among the entity members, that is, which child field member reports to or belongs to which child field member, by selecting both the field and Parent and then clicking .

Removes the selected child or parent field.

Properties
The properties box presents a node model of the org chart. Data fields, labels and images can be inserted into the node as the information about the entity in the org chart, by using . By default all added objects are placed at the left top of the node, you need to adjust their positions and sizes in the node. You can also resize the node if required.

To remove an object from the node, select it and then click .

For heat map

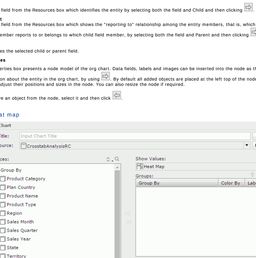


Chart Title
Specifies the title of the chart. The title is a special label bound with the chart. Though it can be positioned freely in a report, once you remove the chart from the report, the title will be removed too.

Specifies the font properties of the chart title.

Data Source
Displays the business view that has been used in the chart.

Filter
Opens the Query Filter dialog to specify the filter you want to apply to the selected business view.

Resources
Displays the resources that can be added to the chart.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

Launches the quick search toolbar to search for view elements.

Adds the selected field into the Groups or Summaries box.

Removes the selected field from the Groups or Summaries box.

Show Values
Displays Heat Map as the selected chart type.

Groups
Lists the fields used to group the data. There should be at least one group.

- **Color by**
  Specifies whether to color by a group. 0-n groups can be used as the color-by fields.
- **Label by**
  Specifies whether to show a group field in the innermost rectangle. 0-n groups can be used as the label-by fields.

Moves the selected group field one level up.

Moves the selected group field one level down.

Opens the Group Options dialog to define the sort order of the group values and specify the number of the group values that will be displayed in the chart.

Removes the selected group field.

Summaries
Lists the summaries used as size-by/color-by or displayed in the innermost rectangle.

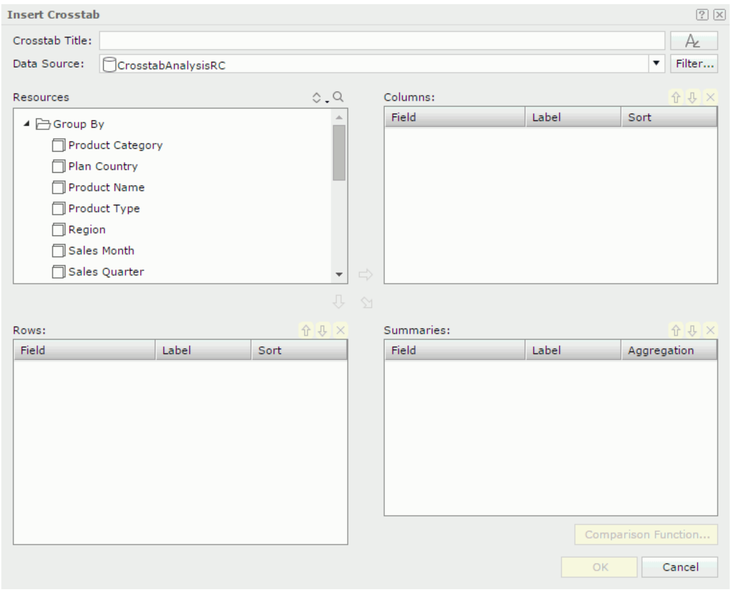
The summaries should match the groups. For example, if the groups level is A > B > C, the static summaries grouped by C can be inserted into the Summaries box, but the static summaries grouped by A, B or other fields cannot.

If no group is specified in the Groups box, you can insert any static summary. And no group-by field will be inserted into the Groups box automatically.

- **Size By**
  Specifies to size by one summary or none.
- **Color By**
  Specifies to color by one summary or none.
- **Label by**
  Specifies whether to show a summary in the innermost rectangle. 0-n summaries can be used as the label-by fields.

Insert Crosstab dialog

This dialog appears when you drag Crosstab from the Components panel to the destination. It helps you to insert a crosstab to a report.



Crosstab Title

Specifies a title for the crosstab.

Specifies the font properties of the crosstab title.

- **Font**
  Lists all the available font faces that can be selected to apply to the title.
- **Font Style**
  Specifies the font style of the title. It can be one of the following: plain, bold, italic, and bold italic.
- **Size**
  Specifies the font size of the title.
- **Align**
  Specifies the position of the title to be left, right, center or justify.
- **Font Color**
  Specifies the font color of the title.
- **Background Color**
  Specifies the background color of the title.

Data Source

Specifies a data source on which the crosstab will be built from the data source drop-down list.

Filter

Opens the Query Filter dialog to specify the filter which you want to apply to the selected business view.

Resources

Displays the elements in the selected business view.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.
-
  Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
-
  Highlights the previous matched text.
-
  Highlights the next matched text.

Adds the selected group object to be displayed in the columns of the crosstab.

Adds the selected group object to be displayed in the rows of the crosstab.

Adds the selected aggregation object or detail object to be the summary field of the crosstab.

Columns/Rows

- **Field**
  Lists the group objects that will be displayed in the columns/rows of the crosstab.
- **Label**
  Specifies the display names for the selected group objects.
- **Sort**
  Specifies the sort order of the group objects.

Summaries

- **Field**
  Lists the fields that you select to create summaries.
- **Label**
  Specifies the display names for the selected fields.
- **Aggregation**
  Specifies the functions used to summarize data of the selected detail objects.
- **Comparison Function**
  Opens the Comparison Function dialog to add a comparison function as an aggregate for the crosstab.

Moves the selected view element one level up.

Moves the selected view element one level down.

Removes the selected resource.

OK

Inserts a crosstab and closes the dialog.

Cancel

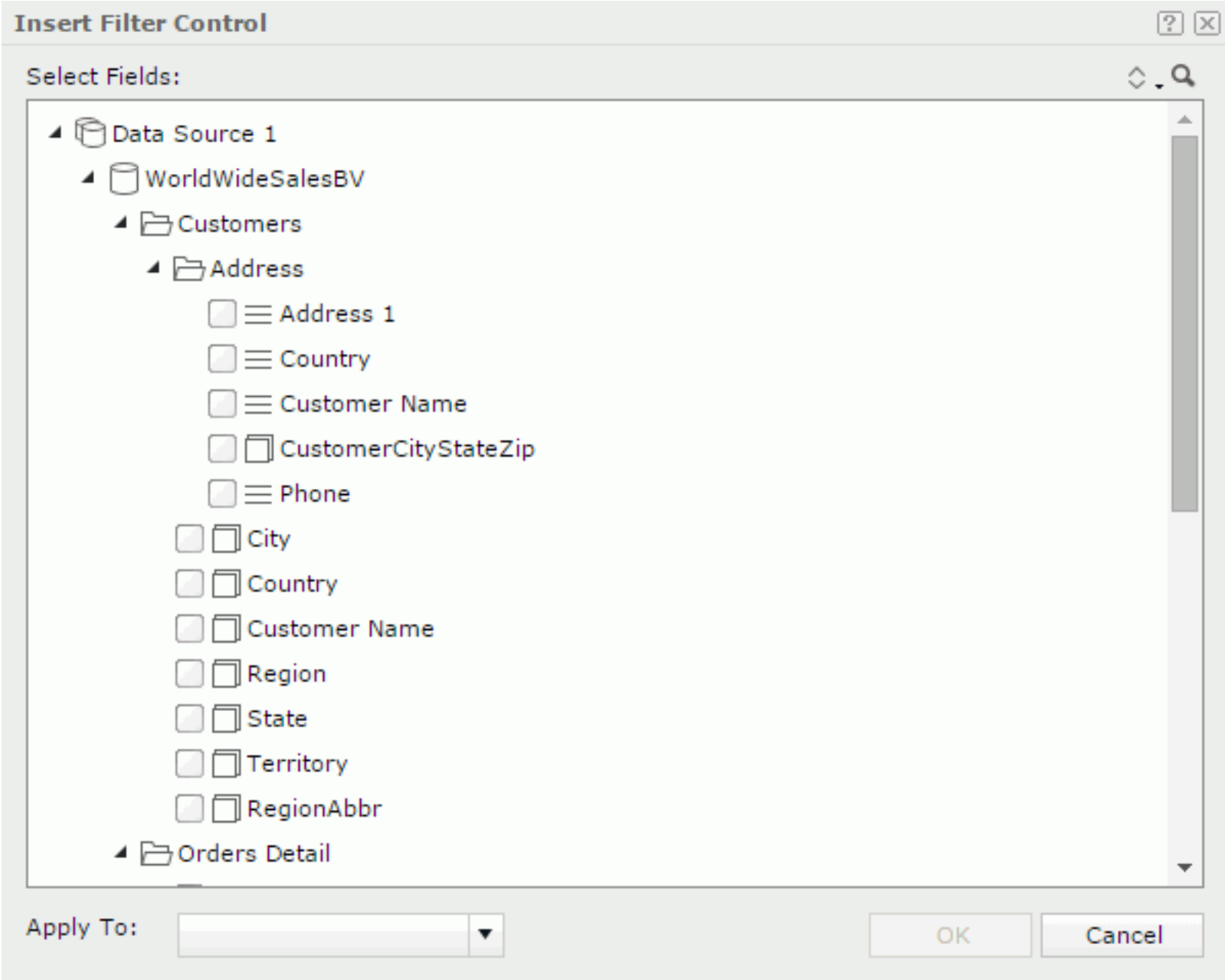
Cancels the insertion and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Insert Filter Control dialog

The dialog appears when you drag Filter Control from the Components panel to a report. It helps you to insert a filter control into a report for filtering data components in the report.



Select Fields

Specifies the fields to bind to the filter control. All the selected fields should be of the same data type. All uncomparable data type fields cannot be bound to filter controls, such as Binary, Blob, Clob, Longvarchar, Longvarbinary, and Varbinary.

The common usage is to select one field, and then based on the field to filter the data of the components created from the same data source as the field.

To filter components using different data sources, choose a common field all the data sources contain and select the field in all the data sources.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.
-
  Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
-
  Highlights the previous matched text.
-
  Highlights the next matched text.

Apply To

Specifies the components to which the filter created with the filter control will be applied.

OK

Inserts a filter control into the report and closes the dialog.

Cancel

Cancels the insertion and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Insert Image dialog

This dialog appears when you drag Image from the Components panel to a report, or click ... on the Page screen of the Web Report Wizard. It helps you to insert an image into a report.

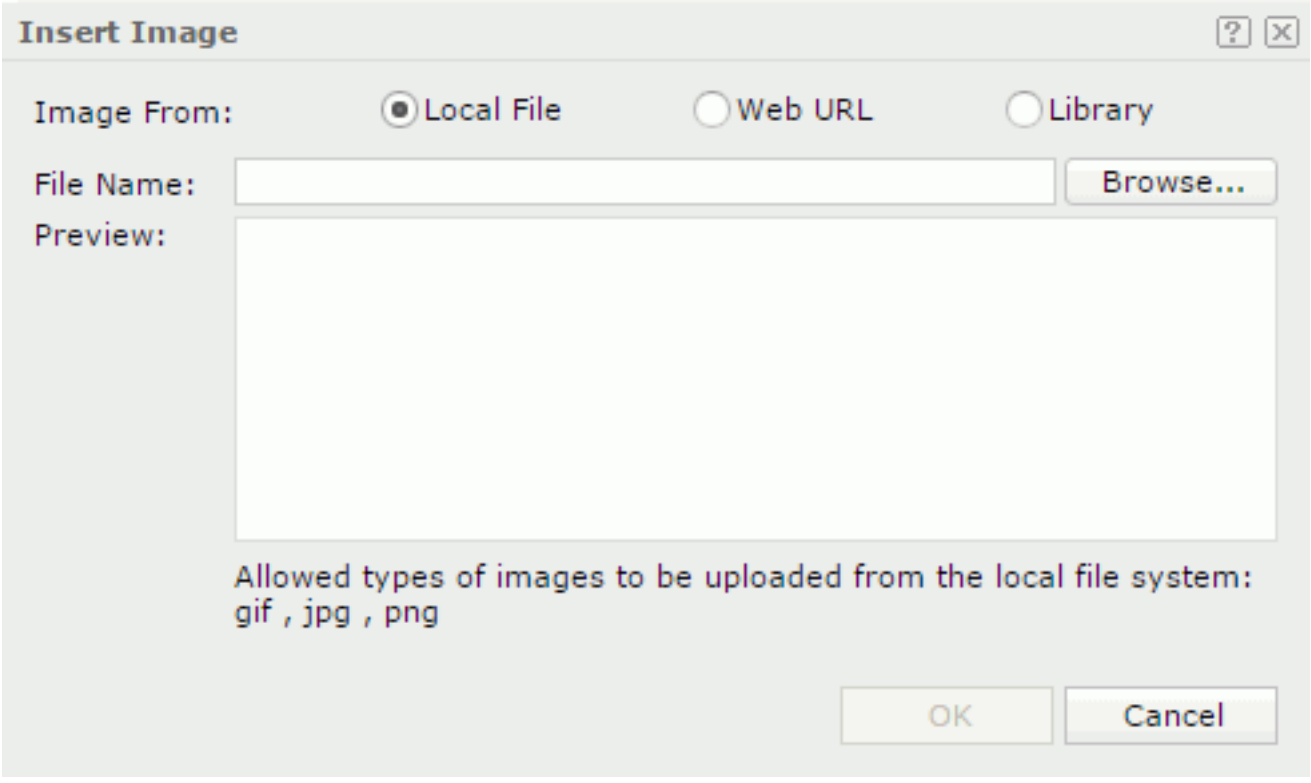


Image From

Specifies the source of the image file.

- **Local File**
  Specifies to use an image from the local file system.
  - **File Name**
    Specifies the path and name of the image file. You can click the Browse button to locate the image file.
- **Web URL**
  Specifies to use an image via URL.
  - **Image URL**
    Specifies the URL of the image file.

    Note: If your JReport Server is in an intranet which requires a proxy, to successfully access the image via URL, you need to add the parameters `-Dhttp.proxyHost=XXX -Dhttp.proxyPort=XX` to the server's startup file JRServer.bat, which is located in `<install_root>\bin`.
- **Library**
  Specifies to use an existing image.
  - **My Images**
    The My Images folder is a virtual location where JReport Server stores the images that have once been inserted into reports. Select the one you want to use.

Preview

Displays a preview of the selected image.

OK

Inserts the image into the report and closes the dialog.

Cancel

Cancels the insertion and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Insert Link dialog

This dialog appears when you right-click an object, click Link on the shortcut menu. It helps you to link the object to a report, a location specified by a URL, an e-mail address, or a Blob data type field as required.

Conditional Link

Specifies whether it is a conditional link. If checked, the object can be linked to different targets based on different conditions.

- Adds a new condition in the Edit Conditions dialog.
- Edits the selected condition.
- Removes the selected condition.
- Moves the selected condition one step up.
- Moves the selected condition one step down.
- **Others**
  If checked, you can define to which target the object will be linked when none of the conditions you have specified is met.

Link Type

Specifies the type of the link target for the object if it is not a conditional link or for the object under the selected condition if it is a conditional link. It can be one of the following:

- Report
- URL
- E-mail
- Content

More

Displays more link options. Available only for the Report and Content link types.

OK

Applies the settings and closes the dialog.

Cancel

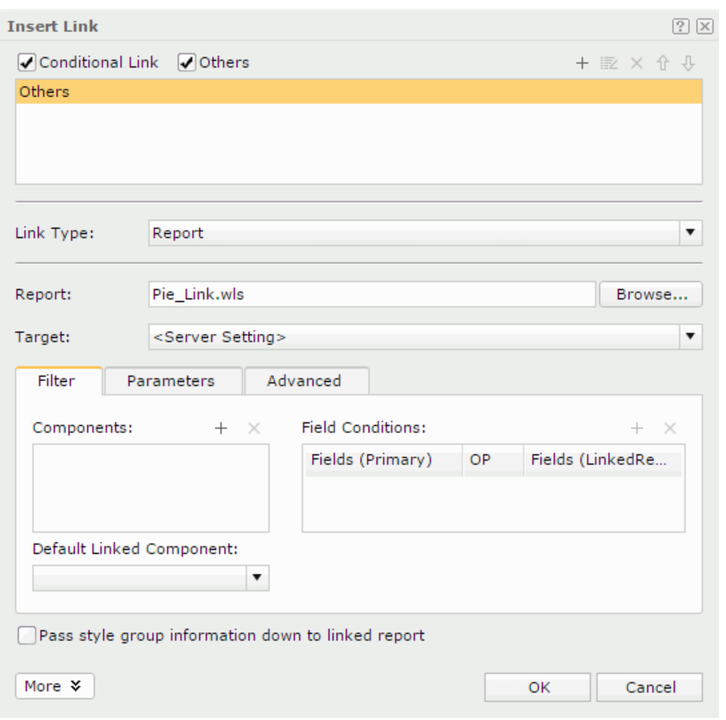
Cancels the changes and exits the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Link to Report

It helps you to link a specified object to a report.



Report

Specifies the target linked report. Click the Browse button to select the desired report in the Select a Report dialog.

Target

Specifies the window or frame in which to load the linked report.

- **New Window**
  Loads the linked report into a new window. This window is not named.
- **Parent Frame**
  Loads the linked report into the parent frame of the frame that contains the link. Not available for link in chart.
- **Same Frame**
  Loads the linked report into the same frame as the link.
- **Whole Window**
  Loads the linked report into the full browser window. Not available for link in chart.
- **<Server Setting>**
  Loads the linked report according to setting of the Target of Detail Table Report and Links option in the Customize Profile panel of the Profile dialog.
- **Other Frame**
  Loads the linked report into some other specified frame. If the frame name does not exist, the linked file will be loaded into a new window.

Filter tab

Specifies the filter conditions based on which the link relationship between the primary report and the linked report is set up.

- **Components**
  Specifies the components in the linked report that will be interlinked with the primary report.
  - Adds a component in the linked report to be interlinked with the primary report.
  - Removes the selected component.
- **Default Linked Component**
  Specifies the default linked component in the linked report.
- **Field Conditions**
  Lists the link conditions between the primary report and the linked report.
  - Adds a new condition line.
  - Removes the selected condition line.
  - **Fields(Primary)**
    Lists the selected fields of the primary report.
  - **OP**
    Specifies the operator to compose the condition. It can be "=", "<>", "<", ">", "<=", ">=" or "IN".
  - **Fields(LinkedReport)**
    Lists the fields in the datasets of the linked report which are of the same data type as the selected fields in the primary report.

Parameters tab

This tab is available when the linked report uses parameters. It helps you to assign values to the parameters of the linked report automatically.

- **Parameters**
  Lists all parameters contained in the linked report.
- **Value**
  Lists all the DBFields, formulas, summaries and parameters in the primary report which are of the same data type as the parameters of the linked report. Select the field the value of which you want to assign to parameter of the linked report from the drop-down list.

Advanced tab

Specifies to pass the values of the filter objects such as filter controls and sliders in the primary report to those in the linked report.

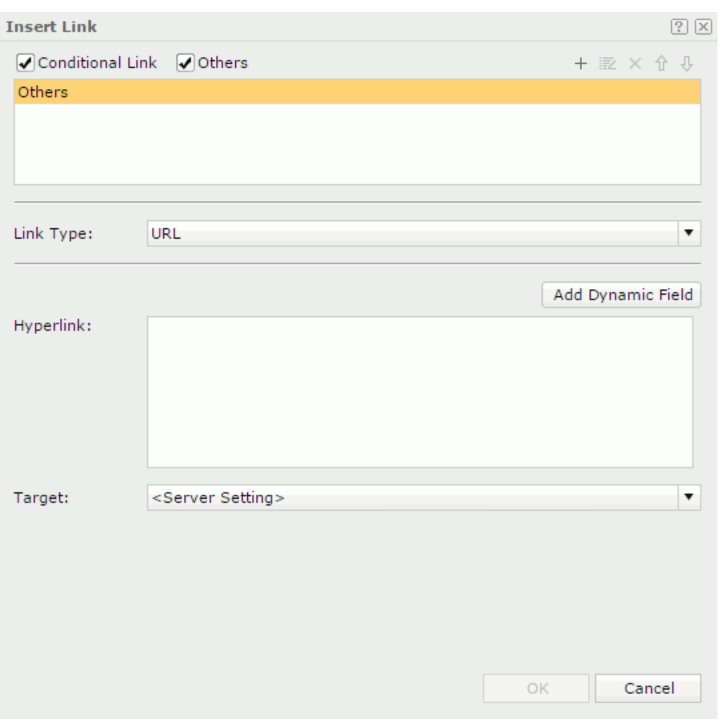
- Adds a new line to pass values between the filter objects in the primary report and the linked report.
- Removes the selected line.
- **Primary Report Property/Object**
  Displays the filter objects in the primary report the values of which will be passed to those in the linked report. Click ... to select a filter object in the primary report in the Select Value dialog.
- **Linked Report Property/Object**
  Displays the filter objects in the linked report which will adopt the values passed from filter objects in the primary report. Click ... to select a filter object in the linked report in the Select Value dialog. You need to make sure that values of the filter object in the primary report can be applied to the filter object in the linked report in the same line.

Pass style group information down to linked report

Specifies whether or not to apply the style group of the primary report to the linked report.

Link to URL

It helps you to link a specified object to a location specified by a URL.



Hyperlink

Specifies the URL for the hyperlink that is to be used to link the object. Fields can work in the hyperlink only when they are inserted via the Add Dynamic Field option.

- **Add Dynamic Field**
  Opens the Select Field dialog to insert a field into the URL.

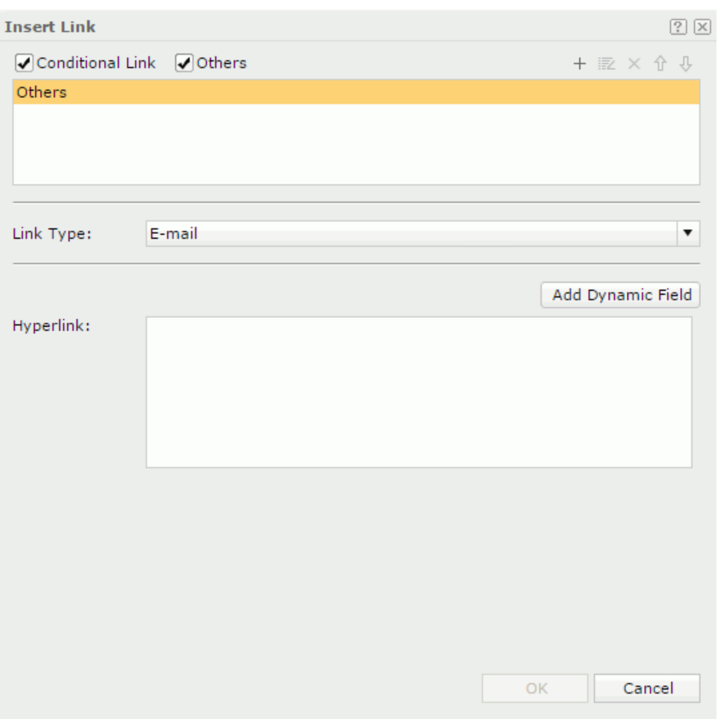
Target

Specifies the window or frame in which to load the location specified by the URL.

- **New Window**
  Loads the linked file into a new window. This window is not named.
- **Parent Frame**
  Loads the linked file into the parent frame of the frame that contains the link.
- **Same Frame**
  Loads the linked file into the same frame as the link.
- **Whole Window**
  Loads the linked file into the full browser window.
- **<Server Setting>**
  Loads the linked file according to setting of the Target of Detail Table Report and Links option in the Customize Profile panel of the Profile dialog.
- **Other Frame**
  Loads the linked file into some other specified frame. If the frame name does not exist, the linked file will be loaded into a new window.

Link to E-mail

It helps you to link the specified object to an e-mail address.



Hyperlink

Specifies the e-mail address that the object will be linked to. Then after you click the trigger object, an e-mail will be popped with the information specified in the Hyperlink box. You can then further customize the e-mail and send it.

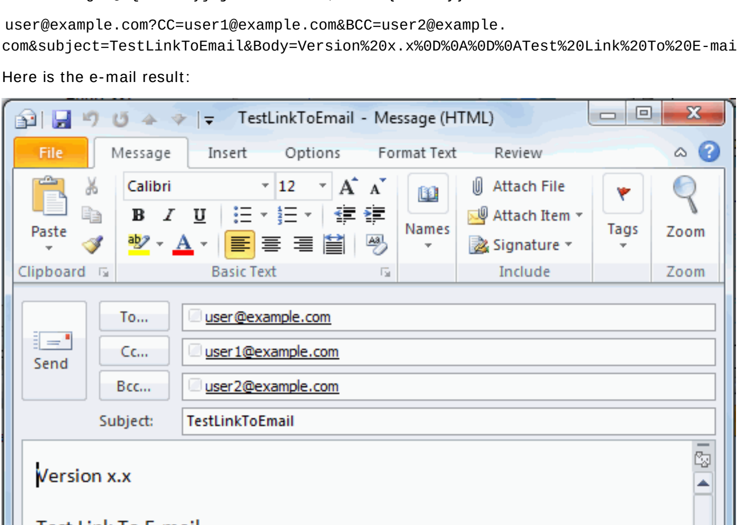
Enter the e-mail address in the Hyperlink box using the syntax `sAddress[sHeaders]`. The following are details about the syntax:

- **sAddress**
  One or more valid e-mail addresses separated by a semicolon. You must use Internet-safe characters, such as %20 for the space character.
- **sHeaders**
  Optional. One or more name-value pairs. The first pair should be prefixed by a "?" and any additional pairs should be prefixed by a "&". The name can be one of the following strings:
  - **CC**
    Addresses to be included in the "cc" (carbon copy) section of the message.
  - **BCC**
    Addresses to be included in the "bcc" (blind carbon copy) section of the message.
  - **subject**
    Text to appear in the subject line of the message.
  - **body**
    Text to appear in the body of the message.

Hyperlink examples:

- `user@example.com`
- `user1@example.com;user2@example.com`
- `${Customer Name}@jinfonet.com`, here `${Customer Name}` is a field name.
- `salesmanager@${Country}.jinfonet.com`, here `${Country}` is a field name.
- `user@example.com?CC=user1@example.com&BCC=user2@example.com&subject=TestLinkToEmail&Body=Version%20x.x%0D%0A%0D%0ATest%20Link%20To%20E-mail`

  Here is the e-mail result:



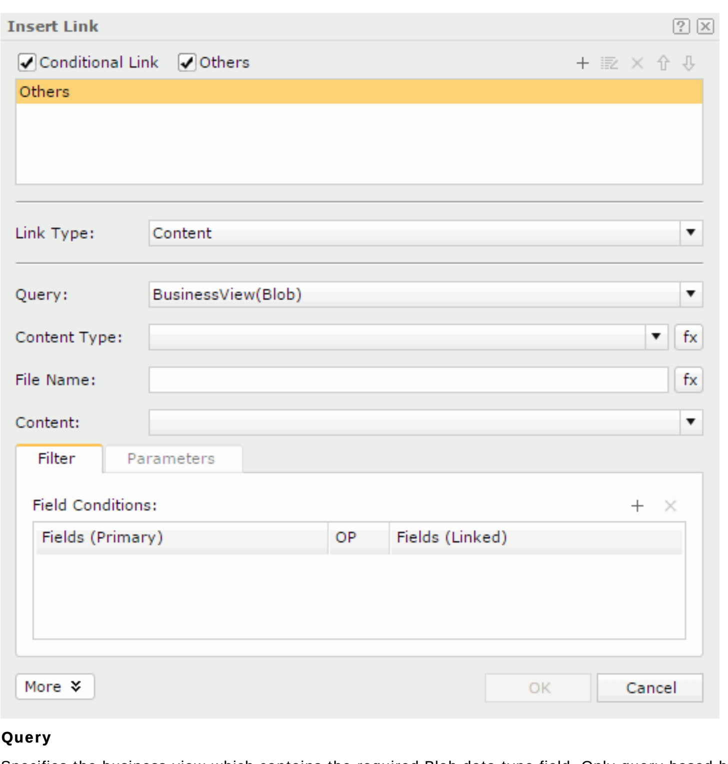
Note: Fields can work in the hyperlink only when they are inserted via the Add Dynamic Field option.

Add Dynamic Field

Opens the Select Field dialog to insert a field into the e-mail address.

Content

It helps you to link the object to a Blob data type field. The Blob data type field will be bound to the object and displayed as a hyperlink, you can download the Blob content by clicking this hyperlink. In JReport, Blob could be image, Binary, Blob, Clob, LongBlob, LongClob, Varbinary, Longvarbinary, and so on.



Query

Specifies the business view which contains the required Blob data type field. Only query based business views from the same data source connection as the current report are listed.

When you set the link type to Content, JReport will check whether there are such kind of business views available. If not, options for this link type will be disabled.

Content Type

Specifies the content type for the Blob type data. You can click the button fx to bind it with a string type business view element or a record level pass one formula in the selected business view, or with a dynamic formula in the current report.

File Name

Specifies the file name for the linked Blob type data. You can click the button fx to bind it with a string type business view element or a record level pass one formula in the selected business view.

Content

Specifies the Blob data type field in the selected business view.

Filter tab

Specifies the link conditions between the business view used by the current report and the business view that contains the linked Blob content.

- **Field Conditions**
  Lists the filter conditions between the two business views.
  - **Fields(Primary)**
    Lists the selected fields of the business view used by the current report.
  - **OP**
    Specifies the operator to compose the condition. It can be "=", "<>", "<", ">", "<=", ">=" or "IN".
  - **Fields(Linked)**
    Lists the DBFields in the datasets of the business view that contains the linked Blob content, which are of the same data type as the selected fields in business view used by the current report.
  - Adds a new condition line.
  - Removes the selected condition line.

Parameters tab

This tab is available when the business view that contains the linked Blob content uses parameters. It helps you to assign values to the parameters automatically.

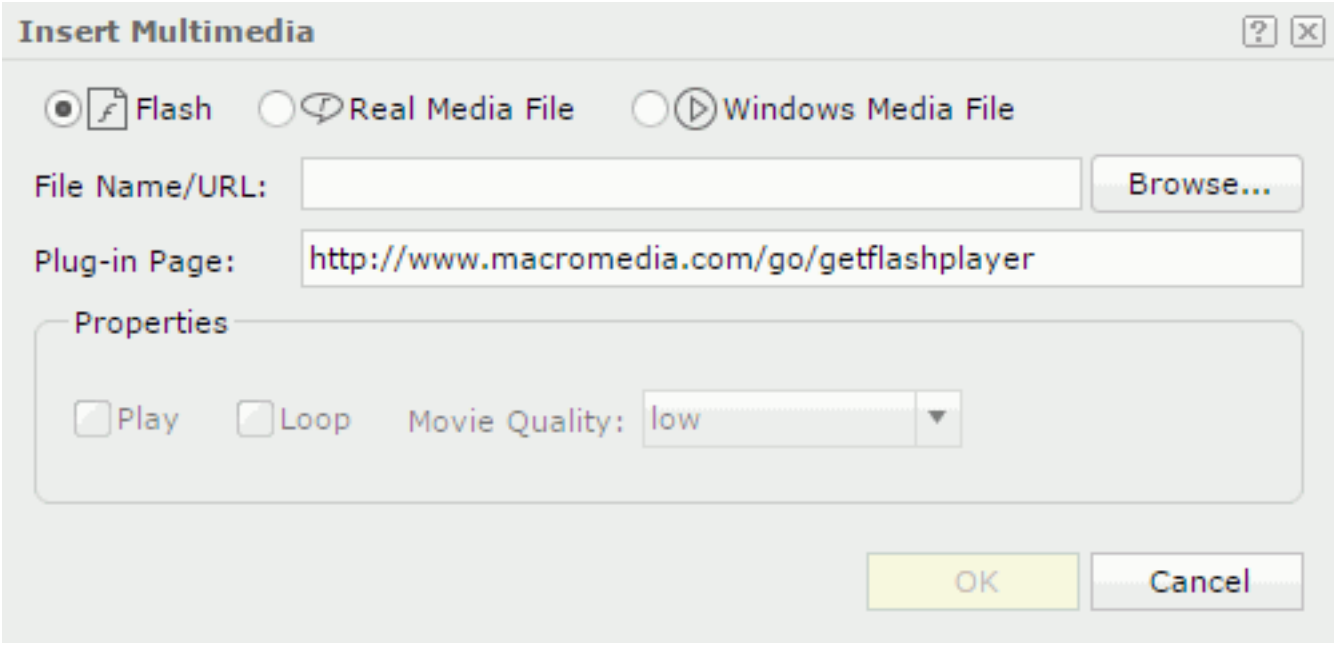
- **Parameters**
  Lists all parameters the business view that contains the linked Blob content uses.
- **Value**
  Lists all the DBFields, formulas, summaries and parameters in the business view used by the current report, which are of the same data type as the parameters the business view that contains the linked Blob content uses. Select the field the value of which you want to assign to a parameter from the drop-down list.

Related topics:

- Binding links to objects

Insert Multimedia dialog

This dialog appears when you drag Multimedia Object from the Components panel to the report. It helps you to insert a multimedia object to a report.



Flash

Specifies to insert a flash file to the current report.

Real Media File

Specifies to insert a real media file to the current report.

Windows Media File

Specifies to insert a Windows Media file to the current report.

File Name / URL

Specifies the name or URL of the multimedia object that is to be inserted into the report. Type in the name or URL in the text box or click Browse to find the file.

Plug-in Page

Specifies the URL of the plug-in page from which to download the player with which to play the inserted multimedia object on a webpage, if the player isn't already installed on your local disk.

Properties

Specifies the properties of the multimedia object.

- **For Flash**
  - **Play**
    Specifies whether to start the flash file automatically when the report is opened.
  - **Loop**
    Specifies whether to play the flash file repeatedly.
  - **Movie Quality**
    Specifies the quality of the flash.
- **For Real Media File**
  - **Auto Start**
    Specifies whether to start the realmedia file automatically when the report is opened.
  - **Center**
    Specifies whether to place the realmedia file in the center of the report.
  - **Loop**
    Specifies whether to play the realmedia file repeatedly.
  - **Loop Number**
    Specifies how many times you want to play the realmedia file.
- **For Windows Media File**
  - **Auto Start**
    Specifies whether or not to start the Windows Media automatically when the report is opened.
  - **Auto Rewind**
    Specifies whether or not to rewind the Windows Media file automatically.
  - **Play Count**
    Specifies the number of times you want to play this media file.

OK

Inserts the multimedia object to the report and closes the dialog.

Cancel

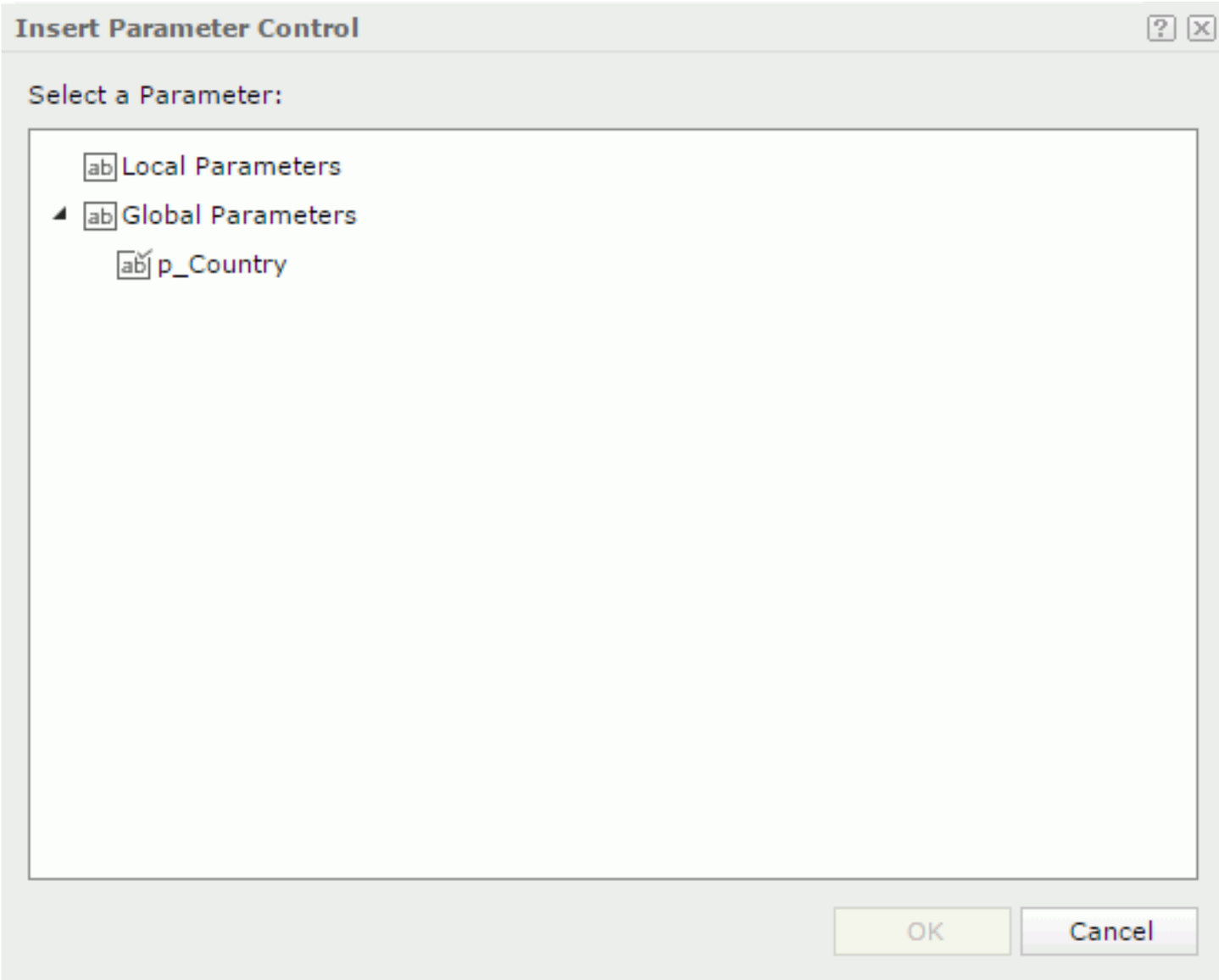
Cancels the insertion and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Insert Parameter Control dialog

The dialog appears when you drag Parameter Control from the Components panel to a report. It helps you to insert a parameter control into a report for delivering parameter values to the report.



Select a Parameter

Specifies a parameter from the list where all the parameters except cascading parameters used by the current report are displayed.

OK

Inserts a parameter control into the report and closes the dialog.

Cancel

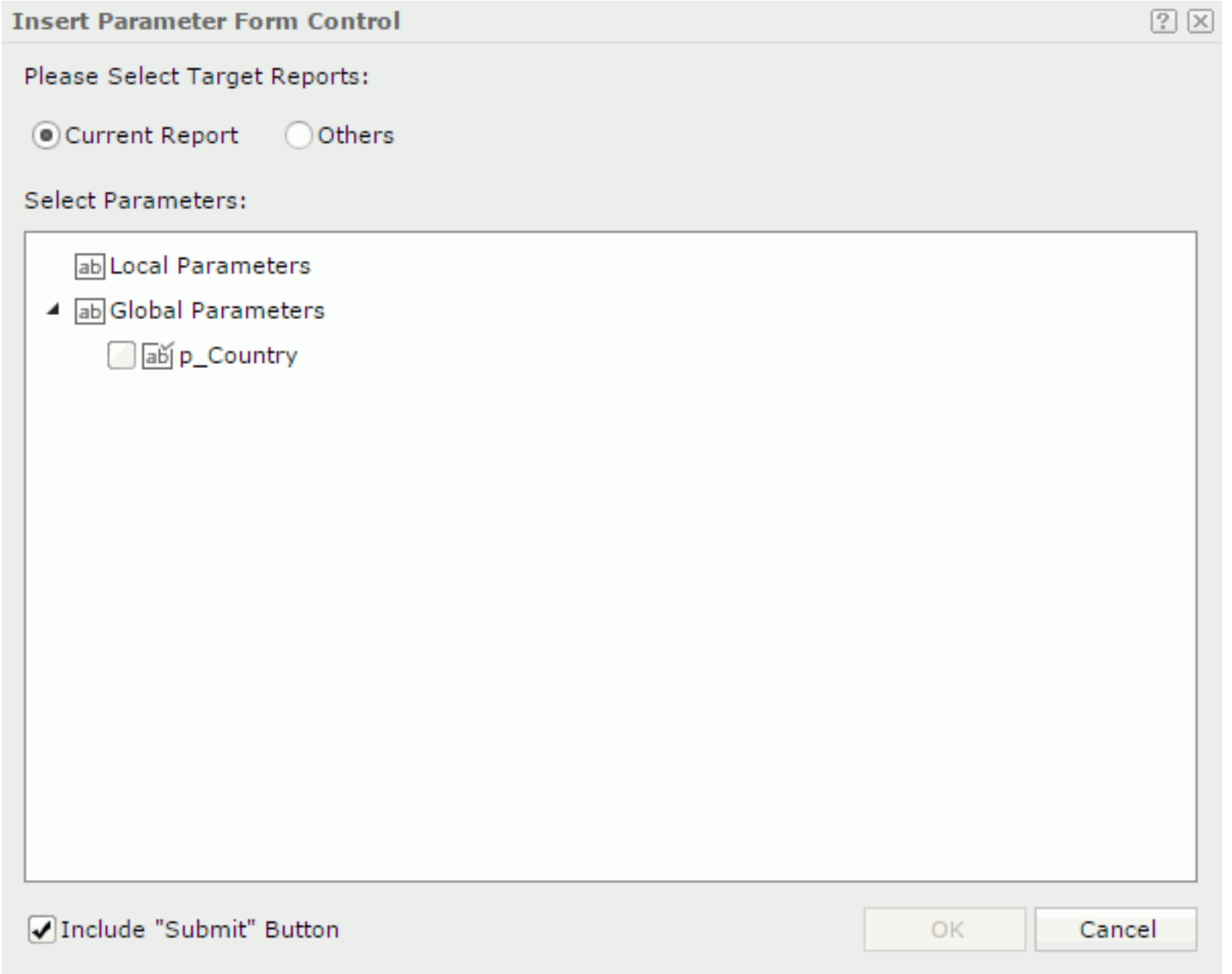
Cancels the operation and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Insert Parameter Form Control dialog

The dialog appears when you drag Parameter Form Control from the Components panel to a report. It helps you to insert a parameter form control into a report for running reports from the current report.



Please Select Target Reports

Specifies the reports to run.

- **Current Report**
  Specifies to run the current report with the specified parameters.
  - **Select Parameters**
    Specifies the parameters that are used to run the current report. Once you select one of a group of cascading parameters, all the other parameters in the cascading group will be selected automatically, and vice versa.
- **Others**
  Specifies to run other reports.
  - **Reports**
    Specifies the target reports to run. If all the selected reports contain no parameters, you cannot finish the dialog.
  - **Parameter Information**
    You can click a report and then view its parameters here.

Include "Submit" Button

Specifies whether the inserted parameter form control contains the Submit button. If Submit is included, it is used to submit the parameter values you specified in the parameter form control. If Submit is not included, once you change the values of a parameter in the parameter form control, the new values will be applied automatically.

OK

Inserts a parameter form control into the report and closes the dialog.

Cancel

Cancels the operation and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Insert Table dialog

This dialog appears when you drag Table from the Components panel to the destination. It helps you to insert a table into a report, and consists of the following tabs:

- Display tab
- Group tab
- Summary tab

Table Title

Specifies a title for the table.

Specifies the font properties of the table title.

- **Font**
  Lists all the available font faces that can be selected to apply to the title.
- **Font Style**
  Specifies the font style of the title. It can be one of the following: plain, bold, italic, and bold italic.
- **Size**
  Specifies the font size of the title.
- **Align**
  Specifies the position of the title to be left, right, center or justify.
- **Font Color**
  Specifies the font color of the title.
- **Background Color**
  Specifies the background color of the title.

Data Source

Specifies a data source on which the table will be built from the data source drop-down list.

Filter

Opens the Query Filter dialog to specify the filter which you want to apply to the selected business view.

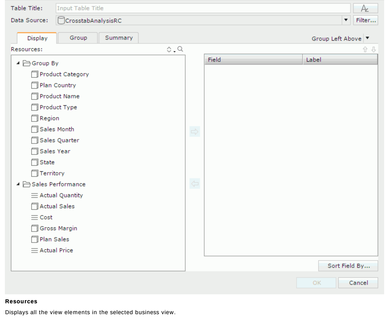
Table Type

Selects a type for the table you want from the Table Type drop-down list. By default, the Group Left Above is selected.

- **Group Above**
  Creates a table with group information above the detail panel.
- **Group Left**
  Creates a table with group information left to the detail panel.
- **Group Left Above**
  Creates a table with group information left above the detail panel.
- **Summary Table**
  Creates a table with only group and summary information.

Display tab

Specifies the fields that you want to display in the table.



Resources

Displays all the view elements in the selected business view.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.
- Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
- Highlights the previous matched text.
- Highlights the next matched text.

Adds the selected view element to be displayed in the table.

Removes the selected view element.

Field

Lists the view elements that have been added to the table.

Label

Specifies the display names for the added view elements.

Moves the selected view element one step up.

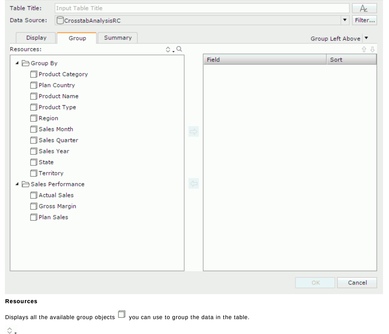
Moves the selected view element one step down.

Sort Fields By

Opens the Custom Sort dialog to specify how to sort data in the table.

Group tab

Specifies the fields to group the data.



Resources

Displays all the available group objects you can use to group the data in the table.

Sorts the group objects in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

Launches the quick search toolbar to search for group objects.

Adds the selected group object as a group field.

Removes the selected group object.

Field

Lists all the group objects that have been added as the group fields.

Sort

Specifies the sort order for groups at the specific group level.

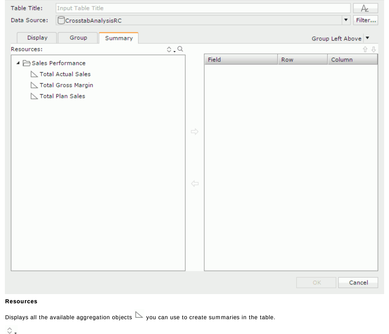
- **Ascend**
  Groups will be sorted in an ascending order (A, B, C).
- **Descend**
  Groups will be sorted in a descending order (C, B, A).
- **No Sort**
  Groups will be sorted in the original order in database.
- **Custom Sort**
  Opens the Custom Sort dialog to set how groups will be sorted.

Moves the selected group one step up.

Moves the selected group one step down.

Summary tab

Specifies the fields on which to create summaries.



Resources

Displays all the available aggregation objects you can use to create summaries in the table.

Sorts the aggregation objects in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

Launches the quick search toolbar to search for aggregation objects.

Adds the selected aggregation object as the summary field.

Removes the selected aggregation object.

Field

Lists the groups that have been added in the table and the aggregation objects added to summarize data in each group.

Moves the selected aggregation object one step up.

Moves the selected aggregation object one step down.

OK

Inserts a table and closes the dialog.

Cancel

Cancels the insertion and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Label Properties dialog

The dialog appears when you right-click a label and select Properties from the shortcut menu. It helps you to specify the properties of the label and contains the following tabs:

- General tab
- Font tab
- Border tab

OK

Applies the settings and closes this dialog.

Cancel

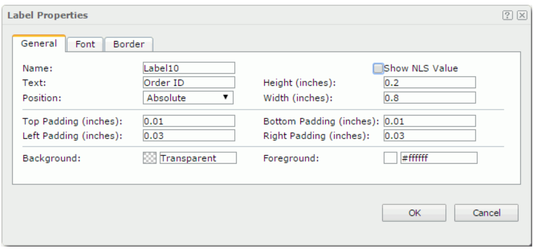
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the label.



Name

Specifies the display name of the label, which will be shown on the shortcut menu of the label.

Show NLS Value

Specifies to show the translated name of the display name of the label in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the label is not modified.

Width

Specifies the width of the label.

Text

Specifies the text of the label.

Height

Specifies the height of the label.

Position

Displays the position mode of the label. If the label is directly contained in the report body, a tabular cell, or a text box, its position mode can be modified.

- **Absolute**: The label's position will be decided by its X and Y property values.
- **Static**: The label will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled.

Top Padding

Specifies the space between the text of the label and its top border.

Bottom Padding

Specifies the space between the text of the label and its bottom border.

Left Padding

Specifies the space between the text of the label and its left border.

Right Padding

Specifies the space between the text of the label and its right border.

Background

Specifies the background color of the label.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

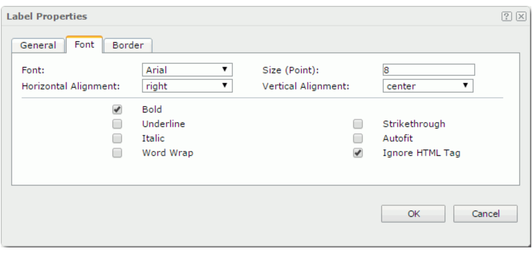
Foreground

Specifies the foreground color of the label.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB.

Font tab

This tab shows the font-related information of the label.



Font

Specifies the font face of the label text.

Size

Specifies the font size of the label text.

Horizontal Alignment

Specifies the horizontal alignment mode of the text in the label.

Vertical Alignment

Specifies the vertical alignment mode of the text in the label.

Bold

Specifies whether to make the label text bold or not.

Underline

Specifies whether the label text will be underlined or not.

Strikethrough

Specifies whether or not to attach a strikeout line to the label text.

Italic

Specifies whether to make the label text italic or not.

Autofit

Specifies whether or not to automatically adjust the width of the label.

Word Wrap

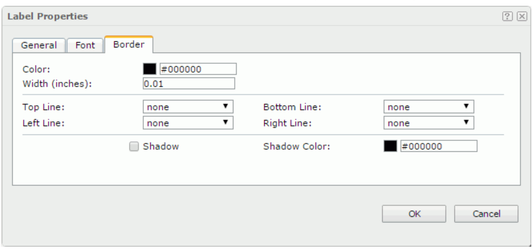
Specifies whether or not to wrap the text to the label width.

Ignore HTML Tag

If this option is unchecked, JReport will parse HTML tag elements in the field value while the report is to be saved as an HTML file; or the field value will appear in the HTML file the same as that in Web Report Studio (HTML tag elements in the field value, if any, will not be parsed).

Border tab

This tab shows information about borders of the label. You can modify all the border settings in this tab.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Shadow

Specifies whether the borders will have a shadow effect or not. This property cannot be rendered in Web Report Studio or JDashboard.

Shadow Color

Specifies the color of the border shadow. This property cannot be rendered in Web Report Studio or JDashboard.

Navigation Control Properties dialog

This dialog appears when you right-click a navigation control and select Properties from the shortcut menu. It helps you to edit the properties of the navigation control and contains the following tabs:

- General tab
- Border tab

OK

Applies the settings and closes this dialog.

Cancel

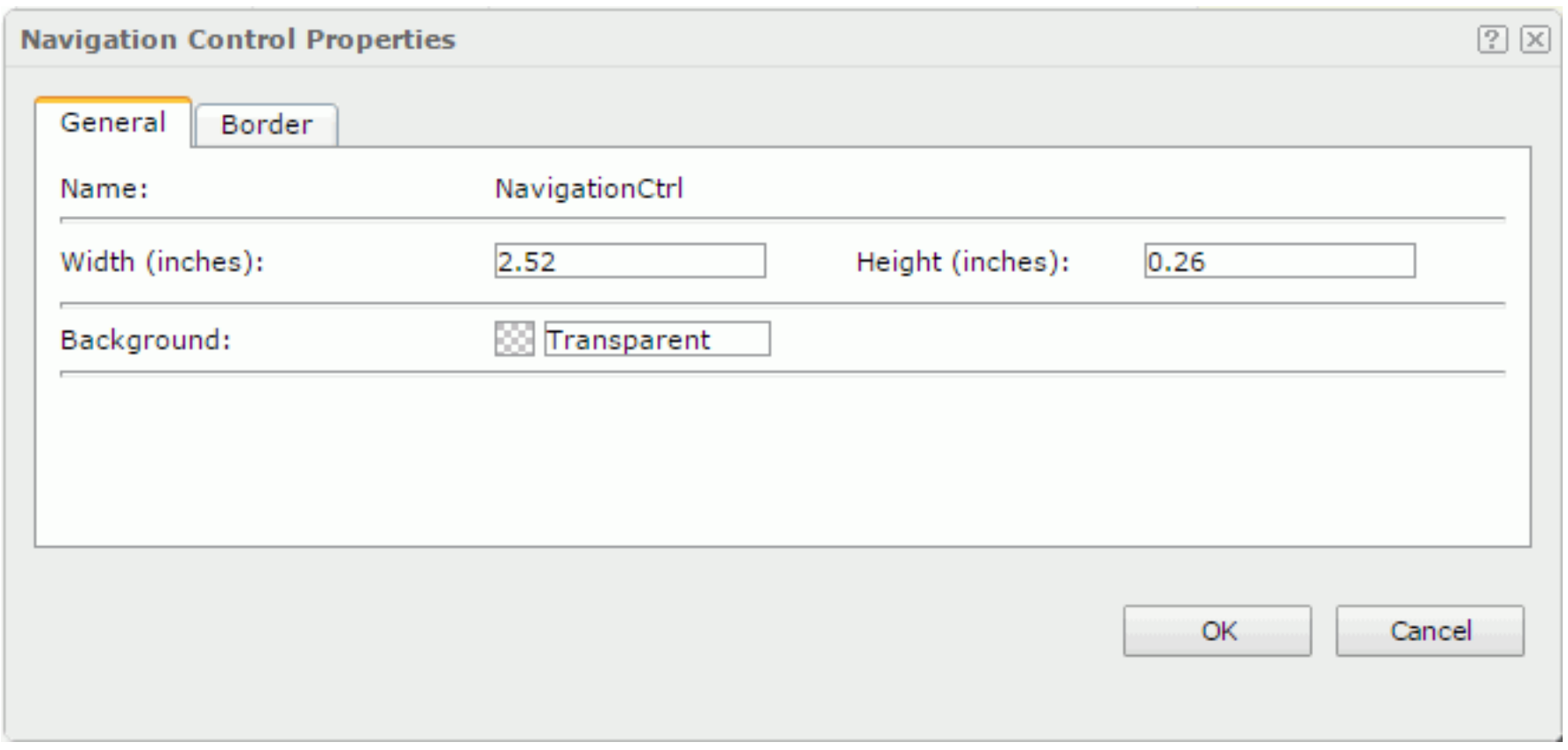
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the navigation control.



Name

Specifies the name of the navigation control.

Width

Specifies the width of the navigation control.

Height

Specifies the height of the navigation control.

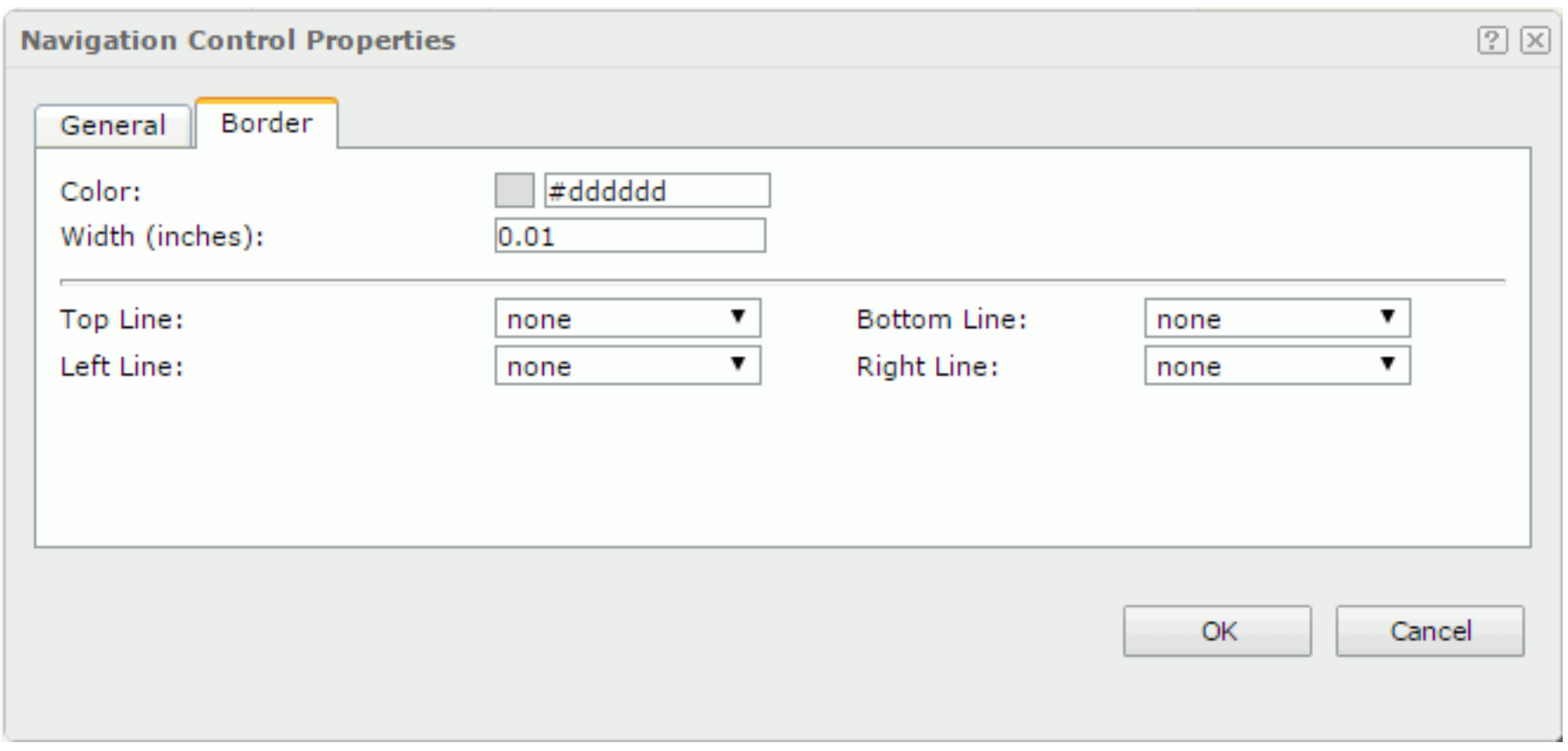
Background

Specifies the background color of the navigation control.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the navigation control.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Page Setup dialog

This dialog varies according to different sources it is opened from.

When you click the Page Setup link in the Web Report Wizard or click Menu > File > Page Setup in Web Report Studio to open the dialog, it helps you to set the page properties of a web report and consists of the following two categories:

- Web Report
- Print Report

OK

Applies all changes and closes the dialog.

Cancel

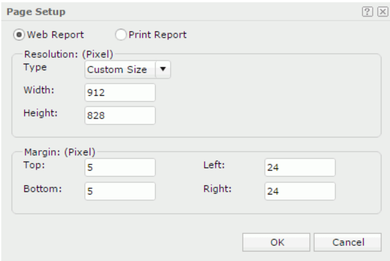
Does not retain any changes and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Web Report

If selected, the unit of page size is pixel, and the dialog appears



Resolution

Specifies the paper resolution. If you select Custom Size, you can then input the paper's Width and Height.

- **Type**
  Specifies the paper type.
- **Width**
  Specifies the paper width.
- **Height**
  Specifies the paper height.

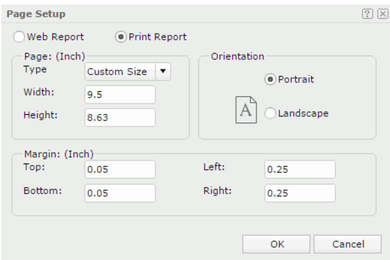
Margin

Specifies the distance of the report data to the top, bottom, left, or right edge of the page.

- **Top**
  Specifies the top margin of the page.
- **Bottom**
  Specifies the bottom margin of the page.
- **Left**
  Specifies the left margin of the page.
- **Right**
  Specifies the right margin of the page.

Print Report

If selected, the unit of the page size is inch, and the dialog appears



Page

Specifies the paper size. If you select Custom Size, you can then input the paper's Width and Height.

- **Type**
  Specifies the paper type.
- **Width**
  Specifies the width of the paper.
- **Height**
  Specifies the height of the paper.

Orientation

Specifies the paper orientation.

- **Portrait**
  The page is positioned vertically.
- **Landscape**
  The page is positioned horizontally.

Margin

Specifies the margin properties.

- **Top**
  Specifies the distance of report data to the top edge of the page.
- **Left**
  Specifies the distance of report data to the left edge of the page.
- **Bottom**
  Specifies the distance of report data to the bottom edge of the page.
- **Right**
  Specifies the distance of report data to the right edge of the page.

When you click the More Settings link in the Print dialog to open the dialog, it helps you to specify the page properties for printing the report result and contains the following options.

Page

Specifies the paper size.

- **Type**
  Specifies the paper type. If you select Custom Size, you can customize the paper's width and height. For any standard type, you can change the width and height too but it will be treated as a custom size then.
- **Width**
  Specifies the width of the paper.
- **Height**
  Specifies the height of the paper.

Margin

Specifies the margin properties.

- **Top**
  Specifies the distance of the layout area to the top edge of the page.
- **Left**
  Specifies the distance of the layout area to the left edge of the page.
- **Bottom**
  Specifies the distance of the layout area to the bottom edge of the page.
- **Right**
  Specifies the distance of the layout area to the right edge of the page.

OK

Applies all changes and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Note: When this dialog is opened from the Print dialog, the setting in the dialog only affects page properties for printing the report result. It doesn't change the current report page setting.

Parameter Control Properties dialog

This dialog appears when you right-click a parameter control and select Properties from the shortcut menu. It helps you to edit the properties of the parameter control and contains the following tabs:

- General tab
- Font tab
- Border tab

OK

Applies the settings and closes this dialog.

Cancel

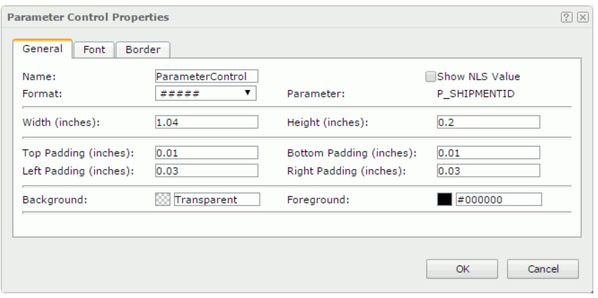
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the parameter control.



Name

Specifies the name of the parameter control.

Show NLS Value

Specifies to show the translated name of the display name of the parameter control in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the parameter control is not modified.

Parameter

Displays the name of the parameter.

Format

Specifies the display format of the parameter value.

Width

Specifies the width of the parameter control.

Height

Specifies the height of the parameter control.

Top Padding

Specifies the space between the text of the parameter control and its top border.

Bottom Padding

Specifies the space between the text of the parameter control and its bottom border.

Left Padding

Specifies the space between the text of the parameter control and its left border.

Right Padding

Specifies the space between the text of the parameter control and its right border.

Background

Specifies the background color of the parameter control.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

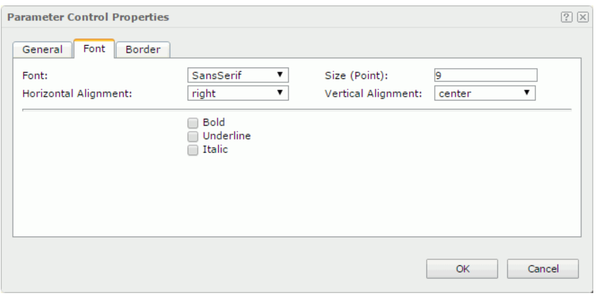
Foreground

Specifies the foreground color of the parameter control.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB.

Font tab

This tab shows the font-related information of the text in the parameter control.



Font

Specifies the font face of the text.

Size

Specifies the font size of the text.

Horizontal Alignment

Specifies the horizontal alignment mode of the text.

Vertical Alignment

Specifies the vertical alignment mode of the text.

Bold

Specifies whether to make the text bold or not.

Underline

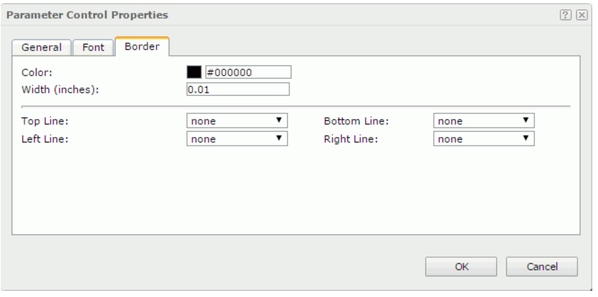
Specifies whether the text will be underlined or not.

Italic

Specifies whether to make the text italic or not.

Border tab

This tab shows information about borders of the parameter control.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Parameter Field Properties dialog

The dialog appears when you right-click a parameter field and select Properties from the shortcut menu. It helps you to modify the properties of the parameter field and contains the following tabs:

- General tab
- Font tab
- Border tab

OK

Applies the settings and closes this dialog.

Cancel

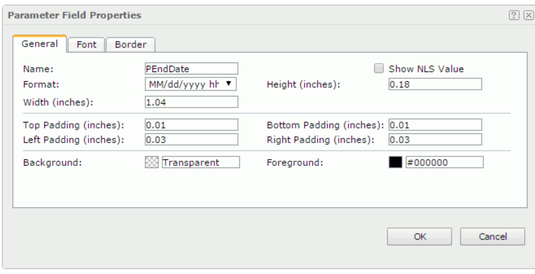
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the parameter field.



Name

Specifies the display name of the parameter field, which will be shown on the shortcut menu of the parameter field.

Show NLS Value

Specifies to show the translated name of the display name of the parameter field in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the parameter field is not modified.

Width

Specifies the width of the parameter field.

Format

Specifies the data format of the parameter field.

Height

Specifies the height of the parameter field.

Top Padding

Specifies the space between the text of the parameter field and its top border.

Bottom Padding

Specifies the space between the text of the parameter field and its bottom border.

Left Padding

Specifies the space between the text of the parameter field and its left border.

Right Padding

Specifies the space between the text of the parameter field and its right border.

Background

Specifies the background color of the parameter field.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

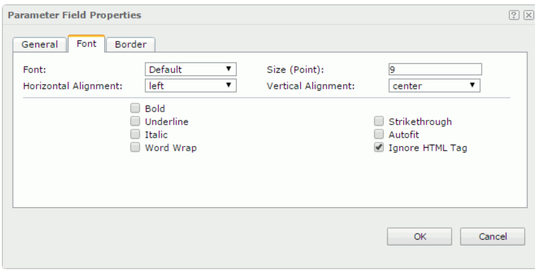
Foreground

Specifies the foreground color of the parameter field.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB.

Font tab

This tab shows the font-related information of the parameter field. You can modify all the font settings in this tab.



Font

Specifies the font face of the parameter field text.

Size

Specifies the font size of the parameter field text.

Horizontal Alignment

Specifies the horizontal alignment mode of the text in the parameter field.

Vertical Alignment

Specifies the vertical alignment mode of the text in the parameter field.

Bold

Specifies whether to make the object text bold or not.

Underline

Specifies whether the field text will be underlined or not.

Strikethrough

Specifies whether or not to attach a strikeout line to the parameter field text.

Italic

Specifies whether to make the field text italic or not.

Autofit

Specifies whether or not to automatically adjust the width of the parameter field.

Word Wrap

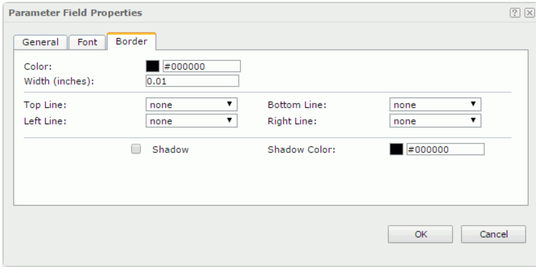
Specifies whether or not to wrap the text to the parameter field width.

Ignore HTML Tag

If this option is unchecked, JReport will parse HTML tag elements in the field value while the report is to be saved as an HTML file; or the field value will appear in the HTML file the same as that in Web Report Studio (HTML tag elements in the field value, if any, will not be parsed).

Border tab

This tab shows information about borders of the parameter field. You can modify all the border settings in this tab.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Shadow

Specifies whether the borders will have a shadow effect or not.

Shadow Color

Specifies the color of the border shadow.

Parameter Form Control Properties dialog

This dialog appears when you right-click a parameter form control and select Properties from the shortcut menu. It helps you to edit the properties of the parameter form control and contains the following tabs:

- General tab
- Border tab
- Title tab

OK

Applies the settings and closes this dialog.

Cancel

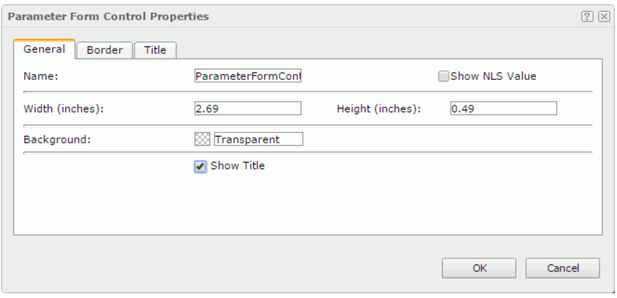
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the parameter form control.



Name

Specifies the name of the parameter form control.

Show NLS Value

Specifies to show the translated name of the display name of the parameter form control in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the parameter form control is not modified.

Width

Specifies the width of the parameter form control.

Height

Specifies the height of the parameter form control.

Background

Specifies the background color of the parameter form control.

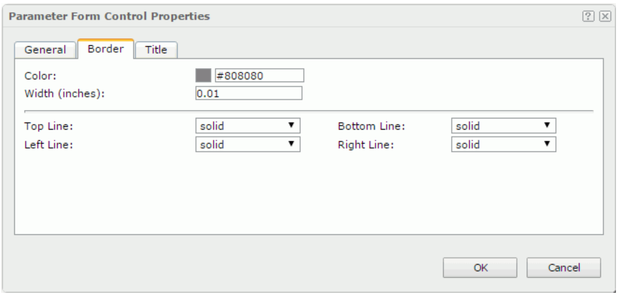
To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Show Title

Specifies whether to show the title of the parameter form control.

Border tab

This tab shows information about borders of the parameter form control.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

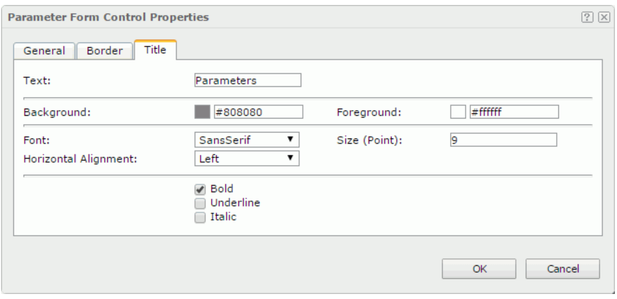
Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Title tab

This tab shows information about the title of the parameter form control. It is available when the Show Title option is selected in the General tab of the same dialog.



Text

Specifies the text of the title.

Background

Specifies the background color of the title.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Foreground

Specifies the foreground color of the title.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB.

Font

Specifies the font face of the text.

Size

Specifies the font size of the text.

Horizontal Alignment

Specifies the horizontal alignment mode of the text.

Bold

Specifies whether to make the text bold or not.

Underline

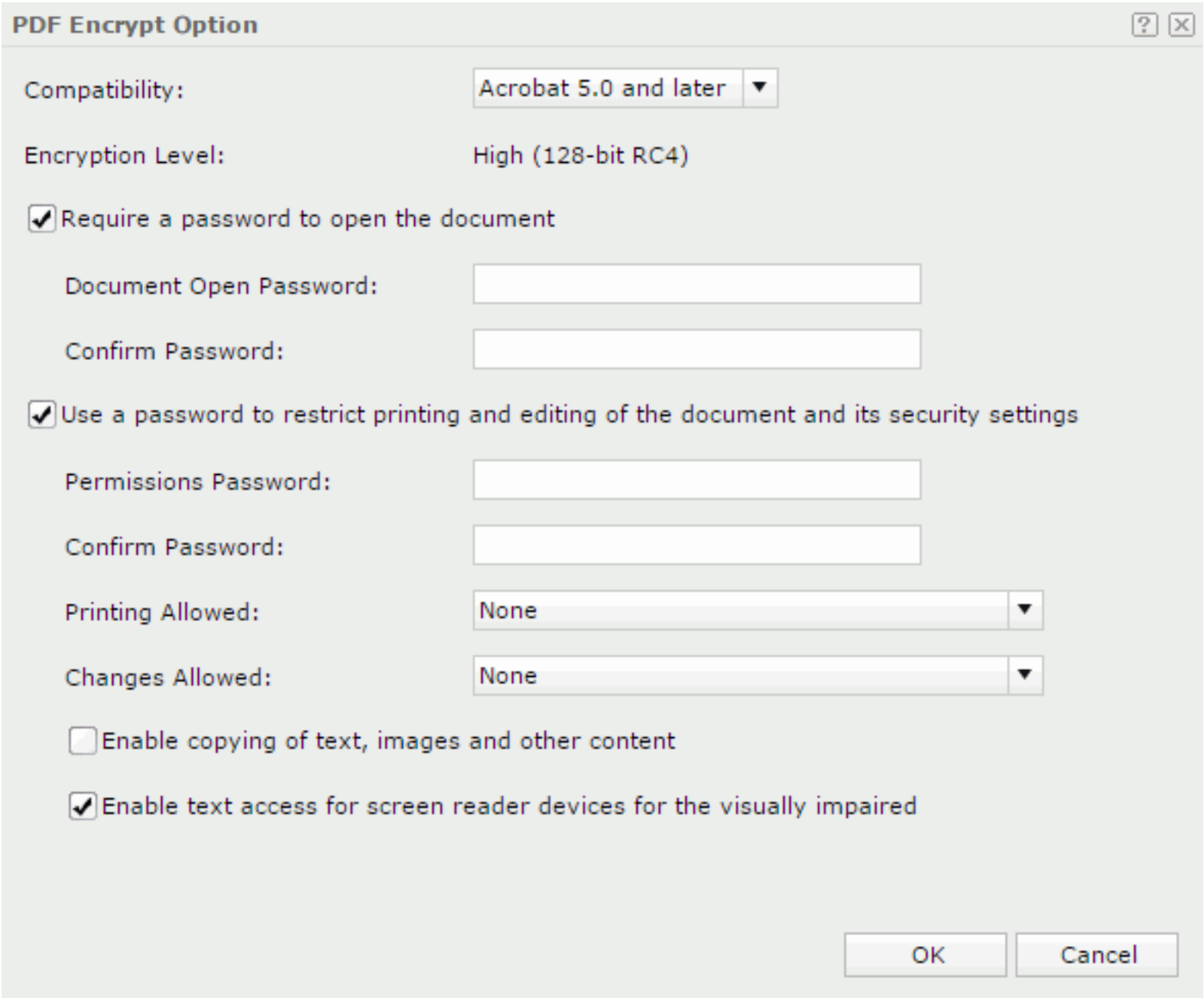
Specifies whether the text will be underlined or not.

Italic

Specifies whether to make the text italic or not.

PDF Encrypt Option dialog

This dialog appears when you check the Encrypt checkbox in the Export dialog. It helps you to set encryption options for the PDF file to be saved.



Compatibility

Specifies the encryption type to encrypt the PDF document. The option Acrobat 3.0 and later uses a low encryption level (40-bit RC4), while the other option Acrobat 5.0 and later uses a high encryption level (128-bit RC4).

Encryption Level

Shows the level of the encryption compatibility that you specified in the Compatibility drop-down list.

Require a password to open the document

Specifies the Document Open Password to prevent others from opening the document without authorization.

- **Document Open Password**
  Specifies the password to prevent others from opening the document without authorization.
- **Confirm Password**
  Confirms the password you have specified in the Document Open Password text box.

Use a password to restrict printing and editing of the document and its security settings

Specifies the Permission Password to prevent others from printing and editing the document. The password you specify here cannot be the same as the one that you use to open the document.

- **Permissions Password**
  Specifies the password to prevent others from printing and editing.
- **Confirm Password**
  Confirms the password you have specified in the Permissions Password text box.
- **Printing Allowed**
  Specifies the printing quality for the PDF document.
- **Changes Allowed**
  Specifies the editing actions that are permitted in the PDF document.

Enable copying of text, images and other content

Allows others to select and copy the contents of the PDF document.

Enable text access for screen reader devices for the visually impaired

Allows visually impaired users to read the PDF document with window readers. This option is available only if the Compatibility option is set to Acrobat 5.0 or later Acrobat version.

OK

Applies the encryption option settings and closes this dialog.

Cancel

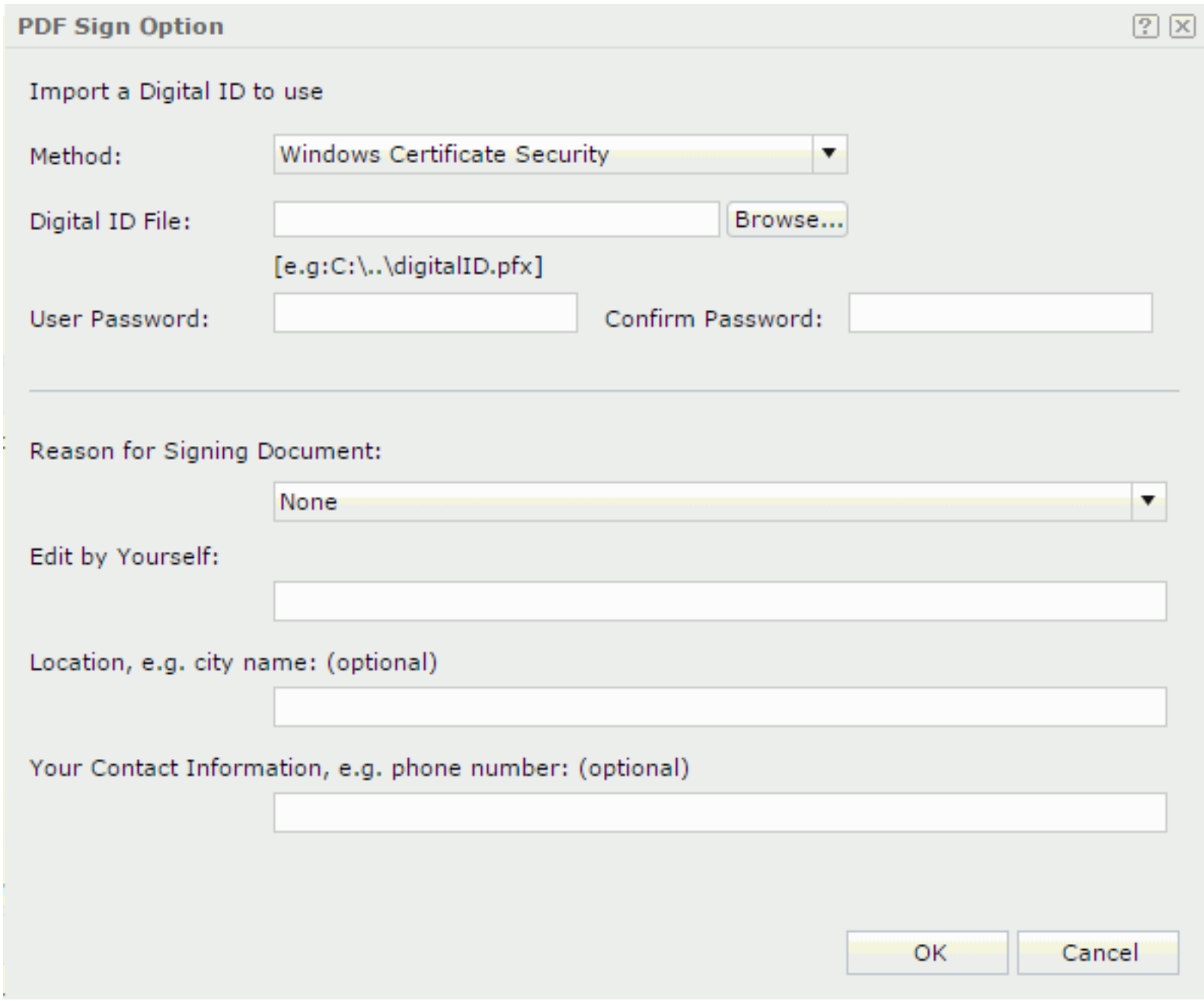
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

PDF Sign Option dialog

This dialog appears when you check the Sign checkbox in the Export dialog. It helps you to set signing options for the PDF file to be saved.



Method

Specifies the digital signing method.

Digital ID File

Specifies whether to use a digital ID file from a specified directory.

User Password

Specifies a password for this digital ID file.

Confirm Password

Confirms the password.

Reason for Signing Document

Specifies the reason for signing the document.

Edit by Yourself

Edits the reason for signing the document. It is enabled when I agree to 'specified' portions of this document is selected from the Reason for Signing Document drop-down list.

Location

Specifies your location accordingly.

Your Contact Information

Specifies your contact information accordingly.

OK

Applies the signing option settings and closes this dialog.

Cancel

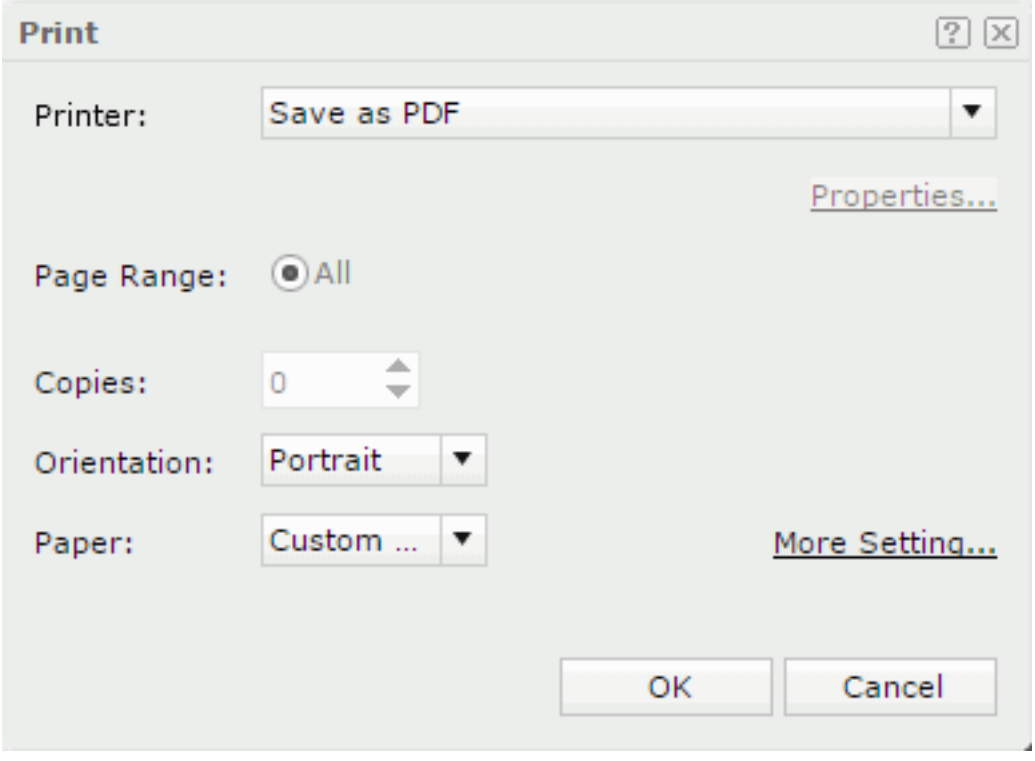
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Print dialog

This dialog appears when you click Menu > File > Print or the Print button on the Standard toolbar in Web Report Studio. It helps you to set the print settings.



Printer

Specifies the printer with which to print the report result. In the Printer drop-down list, all the printers that the web browser can access will be listed, that is to say, the virtual printers displayed in this drop-down list vary according to the operating system and the web browser you are using. If you want to download the result to a PDF or HTML file and then use your local printer to print the file, select Save as PDF or Save as HTML.

Properties

Opens the Printer Properties dialog to specify the printing properties. Disabled when Save as PDF or Save as HTML is selected.

Page Range

Displays the range of the pages that are to be printed.

- **All**
  If checked, all pages will be printed.

Copies

Specifies the number of copies you want to print. The number of copies will be applied to all specified pages. Disabled when Save as PDF or Save as HTML is selected.

Orientation

Specifies the orientation for the printed result.

- **Portrait**
  Prints the result in a standard letter orientation.
- **Landscape**
  Prints the result in a landscape orientation.

Paper

Specifies the type of paper for printing the result. Choose one from the drop-down list.

More Settings

Opens the Page Setup dialog to specify more page properties for the printed result.

OK

Prints the report result with the specified settings. If Save as PDF or Save as HTML is selected in the Printer drop-down list, the result file of the report is then opened in an associated program with which you can print the result to a printer.

Cancel

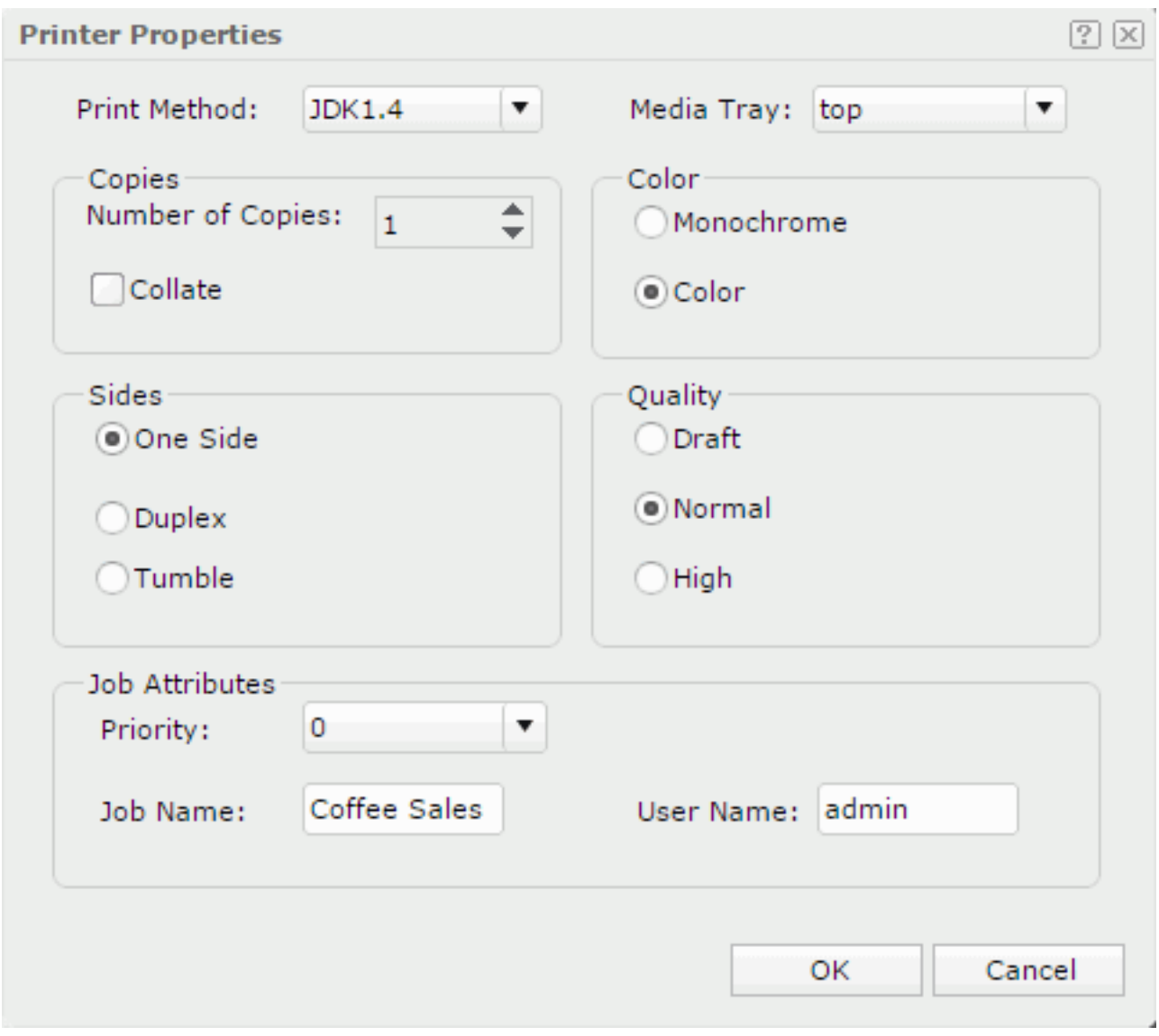
Cancels the print settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Printer Properties dialog

This dialog appears when you click the Properties link in the Print dialog. It helps you to specify the printing properties.



Print Method

Specifies the printing method to print the result.

Media Tray

Specifies the media tray for the printer. Select one from the drop-down list.

Copies

Specifies the information of the copies.

- **Number of Copies**
  Specifies the number of copies you want to print. The number of copies will be applied to all specified pages.
- **Collate**
  Specifies the printing order for printing multiple pages. If checked, when you print two copies of a three-page result, the page order is 1, 2, 3; 1, 2, 3. Otherwise, the order will be 1, 1, 2, 2, 3, 3.

Color

Specifies the color of the result to be printed.

- **Monochrome**
  Specifies to print the result in Monochrome.
- **Color**
  Specifies to print the result in Color.

Sides

Specifies the print sides for the printed result.

- **One Side**
  Imposes each consecutive print-stream page upon the same side of consecutive media sheets.
- **Duplex**
  Imposes each consecutive pair of print-stream pages upon the front and back sides of consecutive media sheets, so that the orientation of each pair of print-stream pages on the medium is correct for the reader as if for binding on the long edge.
- **Tumble**
  Imposes each consecutive pair of print-stream pages upon the front and back sides of consecutive media sheets, so that the orientation of each pair of print-stream pages on the medium is correct for the reader as if for binding on the short edge.

Quality

Specifies the printing quality.

- **Draft**
  Specifies to print the result in draft quality.
- **Normal**
  Specifies to print the result in normal quality.
- **High**
  Specifies to print the result in high quality.

Job Attributes

Specifies the properties of the print task.

- **Priority**
  Specifies the print priority for this task.
- **Job Name**
  Specifies the name of the print task.
- **User Name**
  Specifies the name of the user.

OK

Closes this dialog and applies the setting when printing.

Cancel

Cancels the changes and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Query Filter dialog

This dialog appears when you click the Filter button in the report wizard. It helps you to apply a filter to the business view used by the specified data component to narrow down data scope. The dialog contains the following two modes:

- Basic
- Advanced

Advanced / Basic

Switches the dialog to the advanced/basic mode.

OK

Applies the filter and closes the dialog.

Cancel

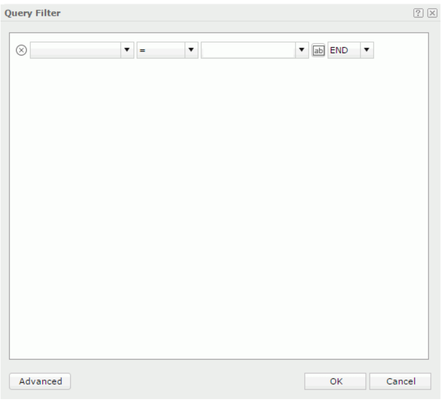
Does not retain changes and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Basic mode

The basic mode provides function for creating simple filter conditions which are connected by AND and OR operators.



Delete the selected condition line.

Field

Specifies the field to be filtered.

Operator

Specifies the operator to compose the filter expression.

- **=**
  Equal to
- **>**
  Greater than
- **>=**
  Greater than or equal to
- **<**
  Less than
- **<=**
  Less than or equal to
- **!=**
  Not equal to
- **[not] in**
  Causes an enumerated list of values to appear in the WHERE clause predicate, used for evaluating for a true condition.
- **[not] like**
  Like string pattern matching operator is used to compare the first expression string value to the pattern string (the second expression). If you want to use wildcard character in the pattern string, only "_" and "%" are supported.
- **[not] between**
  Allows the system to evaluate whether or not data values are located between a range of values indicated in the predicate.
- **is [not] null**
  It is used in WHERE clause predicates to match null values occurring in a specified data field.

Value

Specifies the value of how to filter the field. You can either enter the value manually in the text box or select a value from the drop-down list, or click the button and then select a parameter from the drop-down list as the value. When you type in the value manually, if multiple values are required, they should be separated with ",", and if "," or "\" is contained in the values, write it as "\," or "\\".

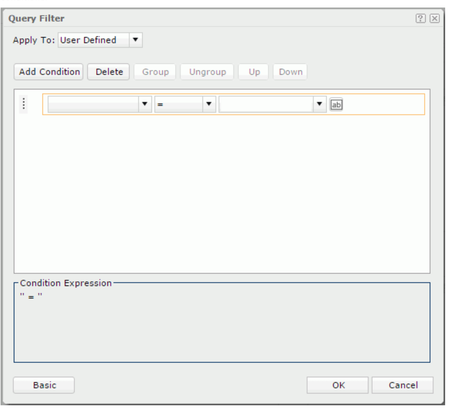
Logic

Lists the logic operator.

- **AND**
  Logic operator AND which is applied to this and the next line.
- **OR**
  Logic operator OR which is applied to this and the next line.

Advanced mode

The advanced mode enables the building of more complex filter conditions via the grouping of conditions.



Filter

Specifies the filter you want to apply to the business view.

- **User Defined**
  Specifies to create a user defined filter to apply to the business view.
- **Predefined filters**
  The names of the predefined filters which were created on the business view in JReport Designer. You can choose one of them to apply.

Add Condition

Adds a new condition line.

Delete

Deletes the selected condition line.

Group

Makes the selected conditions in a group. Conditions can also be added to an existing group by selecting the conditions and the group while holding the Ctrl button, and then clicking the Group button.

Ungroup

Makes the selected group ungrouped.

Up

Moves the selected condition or group up to a higher level.

Down

Moves the selected condition or group down to a lower level.

Logic

Specifies the logic operator.

- **AND**
  Logic operator And which is applied to this and the next line.
- **OR**
  Logic operator Or which is applied to this and the next line.
- **AND NOT**
  Logic operator And Not which is applied to this and the next line.
- **OR NOT**
  Logic operator Or Not which is applied to this and the next line.

Field

Specifies the field to be filtered.

Operator

Specifies the operator to compose the filter expression.

Value

Specifies the value of how to filter the field. You can either enter the value manually in the text box or select a value from the drop-down list, or click the button and then select a parameter from the drop-down list as the value. When you type in the value manually, if multiple values are required, they should be separated with ",", and if "," or "\" is contained in the values, write it as "\," or "\\".

Condition Expression

Displays the SQL statement of the condition.

Related topics:

- Applying filters to business views

RealMedia Properties dialog

The dialog appears when you right-click a RealMedia object and select Properties from the shortcut menu. It helps you to specify the properties of the RealMedia object and contains the following tabs:

- General tab
- Controls tab

OK

Applies the settings and closes this dialog.

Cancel

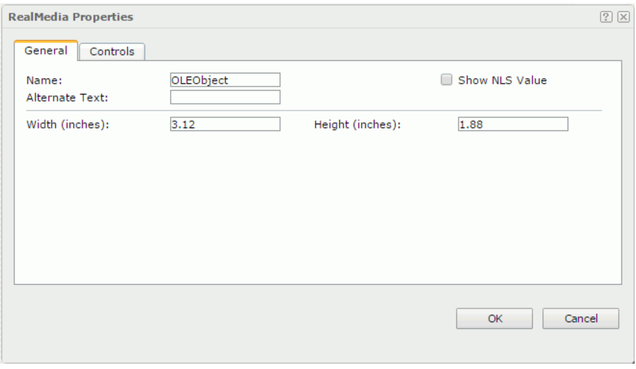
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the RealMedia.



Name

Specifies the display name of the RealMedia.

Show NLS Value

Specifies to show the translated name of the display name of the RealMedia in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the RealMedia is not modified.

Alternate Text

Specifies the alternate text which will be shown if the RealMedia cannot be displayed.

Width

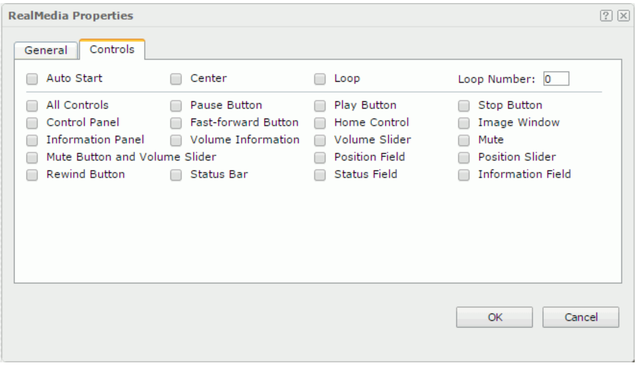
Specifies the width of the RealMedia.

Height

Specifies the height of the RealMedia.

Controls tab

This tab specifies the RealPlayer controls you want to display. The controls will be reproduced in the container as they appear in RealPlayer.



Auto Start

Specifies whether to start the media file automatically when the report is opened.

Center

Specifies whether to place the presentation in the center of the image window and display it in its original size (the size is determined by the Width and Height properties).

Loop

Specifies whether to play the media file repeatedly.

Loop Number

Specifies the number of the times the media file will loop during playback.

Note: If you have checked the Loop option and specified the Loop Number at the same time, the media file will be played back according to the specified number, which means the Loop option will be ignored even though the loop number is set to zero.

All Controls

Specifies whether to show all controls.

Pause Button

Specifies whether to show the Pause Button.

Play Button

Specifies whether to show the Play Button.

Stop Button

Specifies whether to show the Stop Button.

Control Panel

Specifies whether to show the Control Panel, which contains the following playback controls: Play Button, Pause Button, Stop Button, Fast-forward Button, Rewind Button, Position Slider and Home Button.

Fast-forward Button

Specifies whether to show the Fast-forward Button.

Home Control

Specifies whether to show the Home Button, which is connected to the www.real.com website.

Image Window

Specifies whether to show the Image Window, which is used for displaying presentations.

Information Panel

Specifies whether to show the Information Panel that displays the title, author, and copyright for the currently playing clip.

Volume Information

Specifies whether to show the Information/Volume Bar, which consists of the Information Panel and the Mute/ Volume Bar.

Mute Button and Volume Slider

Specifies whether to show the Mute Button and Volume Slider. If checked, the Volume Slider and Mute checkboxes will be checked correspondingly. Otherwise, you can check Volume Slider or Mute respectively.

Position Field

Specifies whether to show the Position Field, which shows the position of the current clip that identifies the clip's current place within the presentation timeline and the total clip length.

Position Slider

Specifies whether to show the Position Slider, which shows the currently playing position within the clip.

Rewind Button

Specifies whether to show the Rewind Button.

Status Bar

Specifies whether to show the Status Bar, which consists of a text message area, the network congestion LED, and the current clip position indicator.

Status Field

Specifies whether to show the Status Field, which consists only of a text message area.

Information Field

Specifies whether to show the Information Field, which displays the title, author, and copyright for the currently playing clip or portion of a multi-clip.

Report Body Properties dialog

The dialog appears when you click Menu > Edit > Report Body Properties. It helps you to specify the properties of the report body and contains the following tabs:

- General tab
- Border tab

OK

Applies the settings and closes this dialog.

Cancel

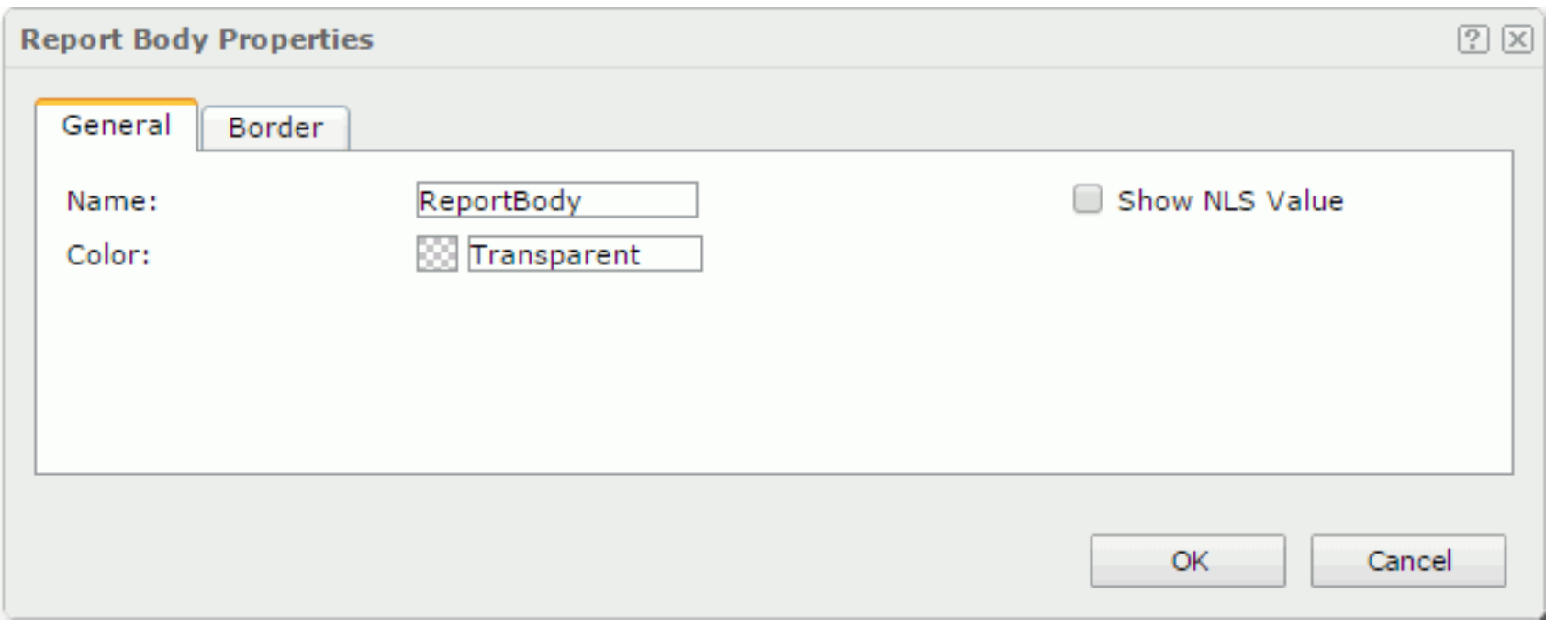
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the report body.



Name

Specifies the display name of the report body, which will be shown on the shortcut menu of the report body.

Show NLS Value

Specifies to show the translated name of the display name of the report body in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the report body is not modified.

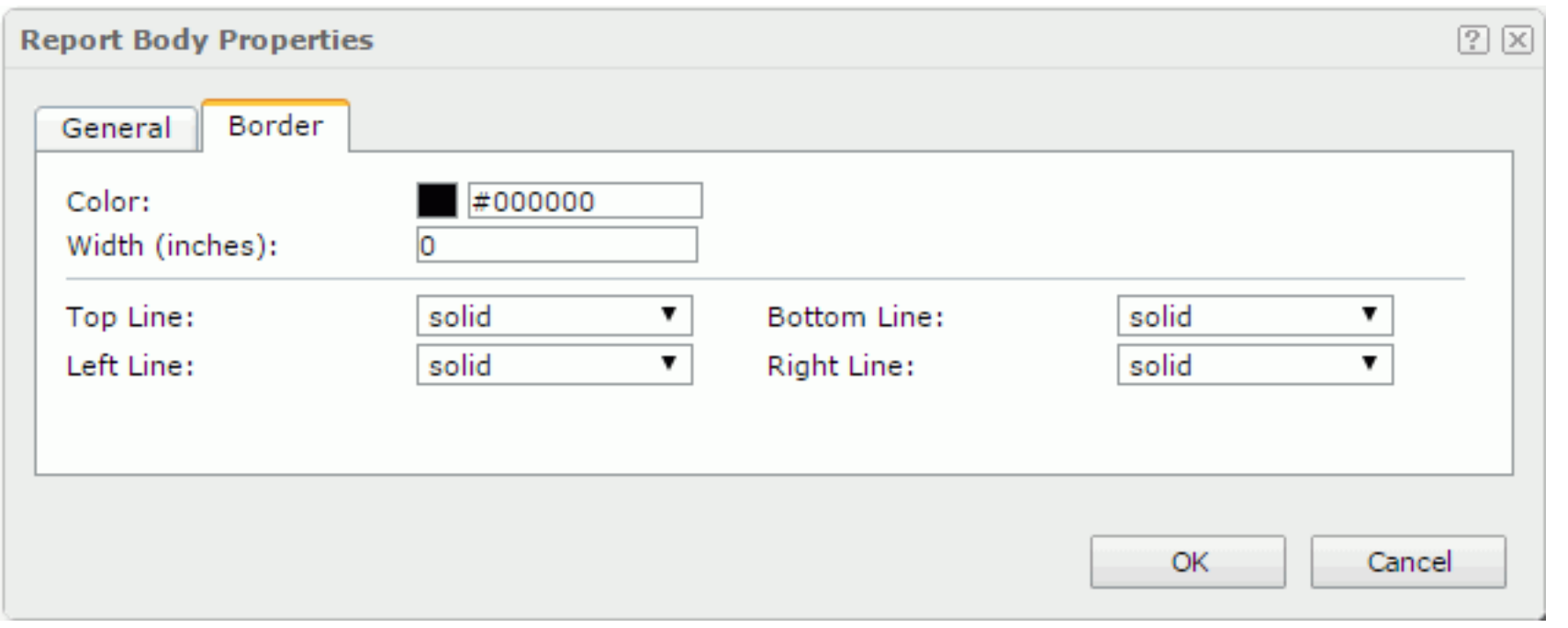
Color

Specifies the color of the report body.

To change the color, click the color indicator to select a color or click **More Colors** in the color indicator to access the Color Picker dialog to specify a new color, or input a color string in the format #RRGGBB. You can also click **More Fill Effects** in the color indicator to specify a gradient or an image as the fill effect of the report body in the Fill Effects dialog. If you want to make the color transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the report body.



Color

Specifies the color of the borders.

Width

Specifies the width of the border.

Top Line

Specifies the line style of the top border of the report body. Choose a style from the drop-down list.

Bottom Line

Specifies the line style of the bottom border of the report body. Choose a style from the drop-down list.

Left Line

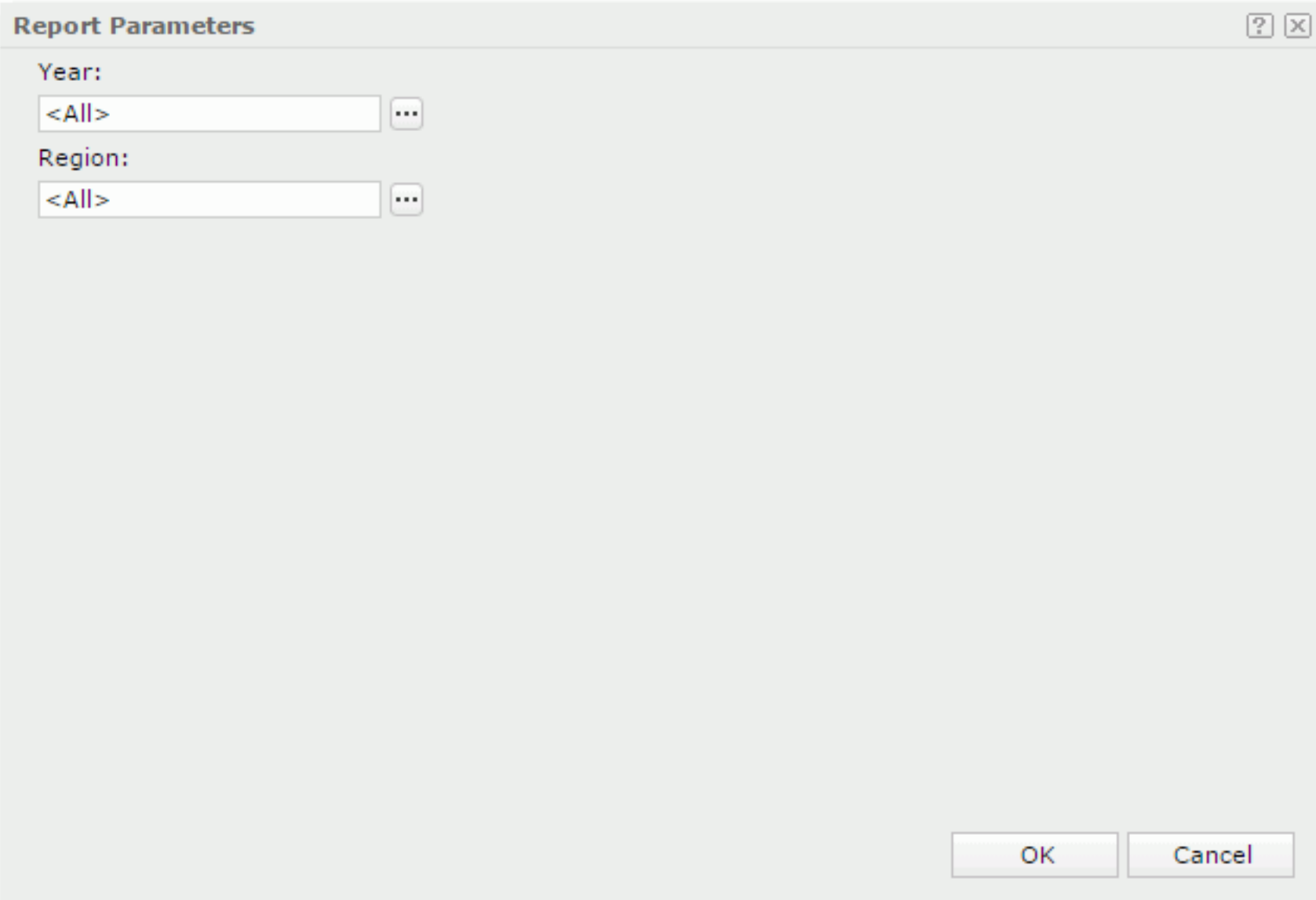
Specifies the line style of the left border of the report body. Choose a style from the drop-down list.

Right Line

Specifies the line style of the right border of the report body. Choose a style from the drop-down list.

Report Parameters dialog

This dialog appears when you click Run in the Web Report Wizard to run a web report with parameters. It helps you to input parameter values for the report.



The way to specify a parameter value varies with the type and properties of the parameter. Here are several ways you can use to specify parameter values:

- In the parameter value combo box, select the required one from the drop-down list or input the value manually.
- Select or unselect the checkbox to specify a Yes/No value.
- Click the button ··· to specify multiple values in the Enter Values dialog.
- Click the calendar button to specify a date and time value using either calendar or expression in the Calendar dialog. If you use an expression to specify the value, when you hover the mouse pointer over this value, a tip will appear showing its expression. Click on the value, the expression will be displayed in the value text box and you can edit the expression in the text box directly if you want. After editing, when you click elsewhere outside of the value text box:
  - If the edited expression is correct, a new value calculated by the expression will be displayed in the parameter value text box.
  - If the edited expression is wrong, no value will be created and the expression itself will be displayed and highlighted in red in the value text box. Correct the expression, or click the calendar button to specify a value.

OK

Accepts the parameter values for further processing.

Cancel

Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Save As dialog

This dialog appears when you click Menu > File > Save As (or the Save As button on the Standard toolbar) or click Save on the Web Report Wizard. It helps you to save the current web report, or save web report template if you are an administrator with the privilege of publishing resources. Options in the dialog vary according to the file type you select.

File Type

Specifies the type of the file. It can be Web Report or Web Report Template. The option Web Report Template is available to administrators with the privilege of publishing resources.

Save

Saves the new file with the specified settings and closes this dialog.

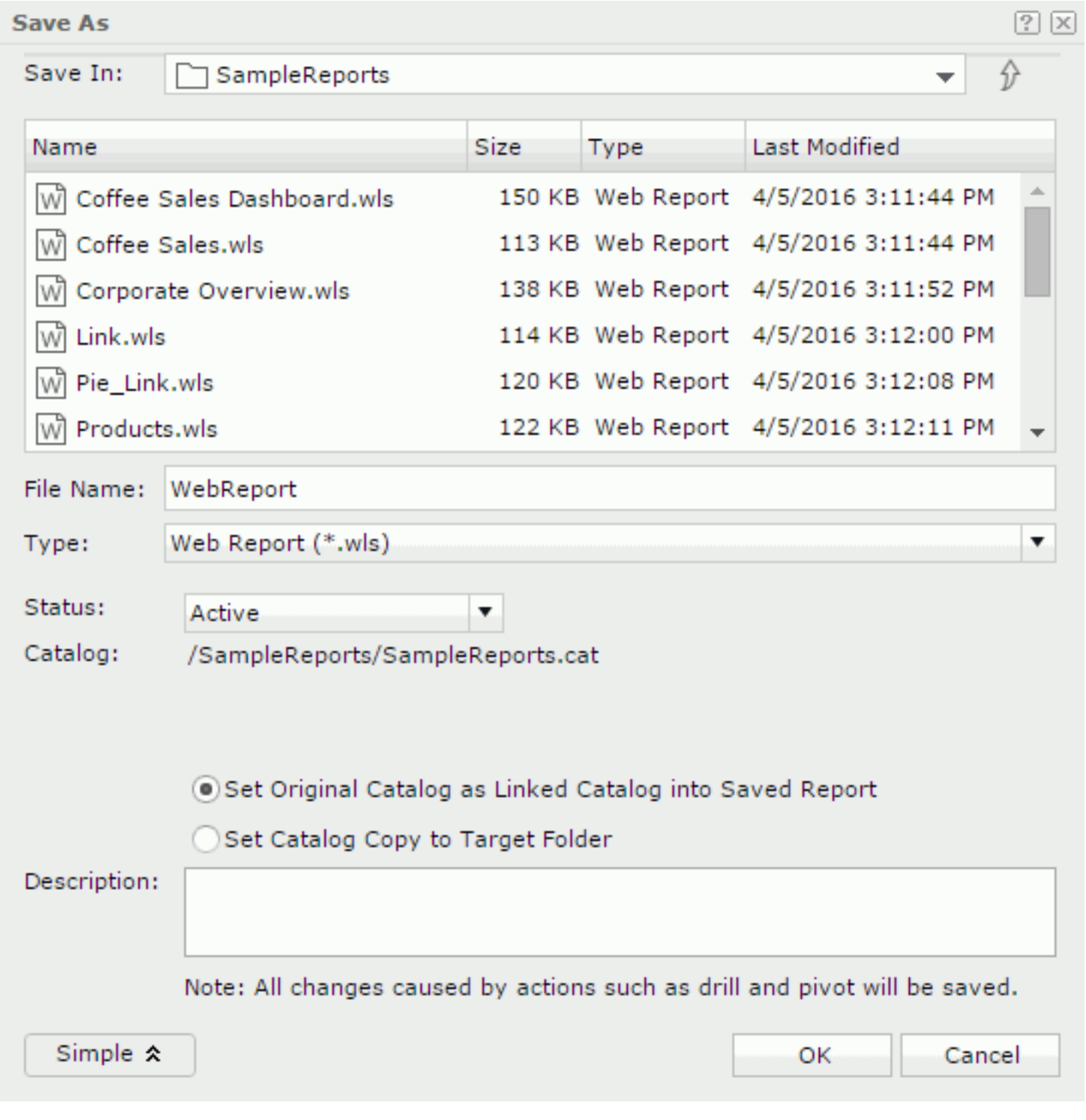
Cancel

Does not save the file as a new one and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

If you select Web Report from the File Type drop-down list to save the current report, the dialog appears



Save in

Specifies the directory in the server resource tree where you want to save the report. The folder may be Public Reports or My Reports. Use the button to go to the parent folder.

The resource table shows the resources in the current directory. Click the column names to change the order of the report in the table list if required.

- **Name**
  Displays the file names.
- **Size**
  Displays the file size.
- **Type**
  Displays the file type.
- **Last Modified**
  Displays the last modified time of the file.

File Name

Specifies the name for the report.

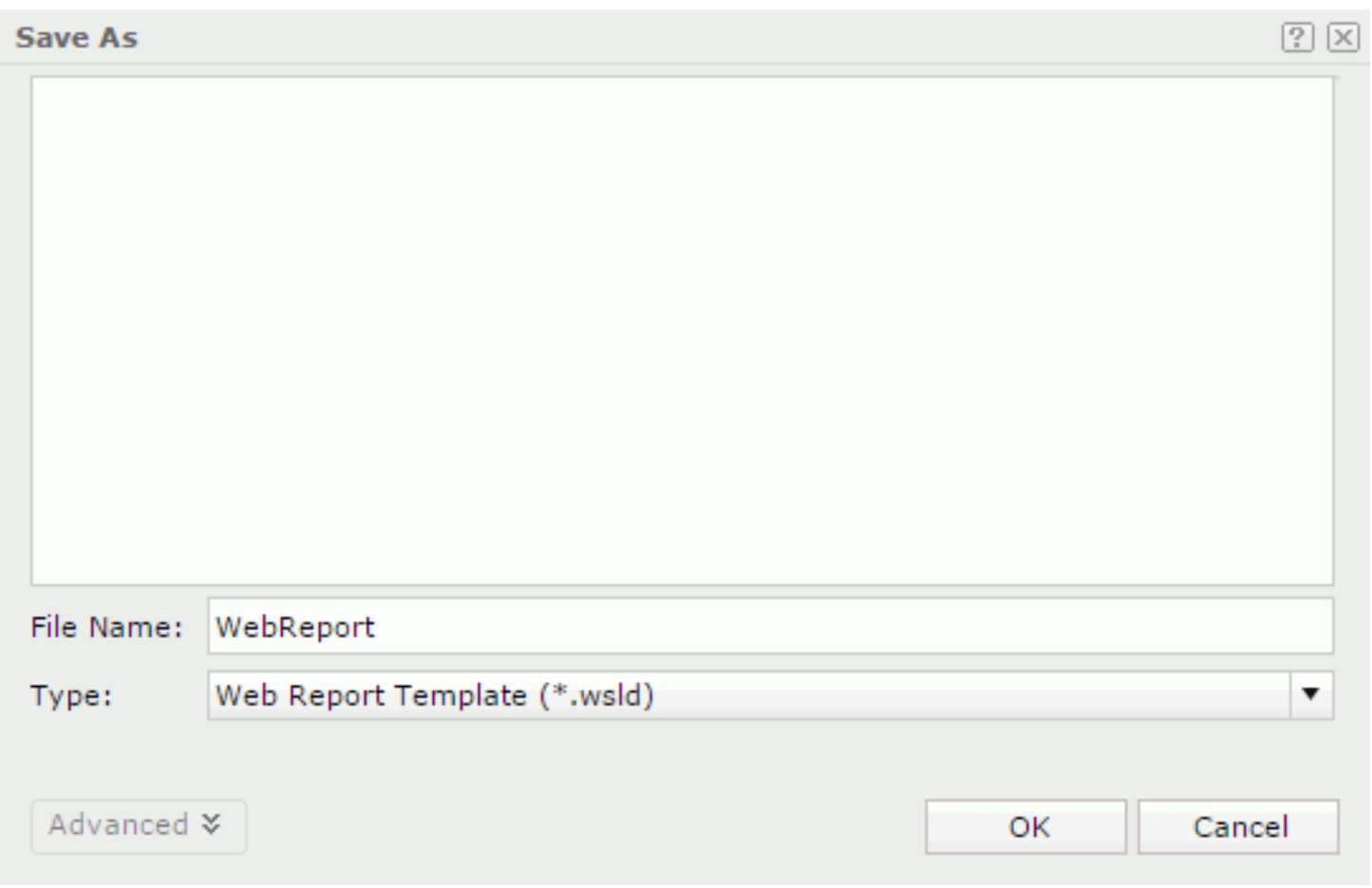
Advanced

Displays the advanced settings.

- **Status**
  Specifies the status of the report.
  - **Active**
    The report can be executed. To execute a report means to run, advanced run and schedule to run it.
  - **Inactive**
    The report cannot be executed. If selected, the corresponding Run, Advanced Run and Schedule commands for the report are not available.
  - **Incomplete**
    The report is not completely designed and cannot be executed. If selected, the corresponding Run, Advanced Run and Schedule commands for the report are not available.
- **Catalog**
  Shows the catalog that the report uses.
  - **Set Original Catalog as Linked Catalog into Saved Report**
    If checked, the saved report will be linked with the catalog and the saved report will run with the catalog no matter whether the two are in the same directory. If later the catalog is updated, the saved report will run with the latest version of the catalog.
  - **Set Catalog Copy to Target Folder**
    If checked, the catalog will be copied to the directory where the report is saved and the saved report will run with the copied catalog.
- **Description**

  Specifies the description for the new file.

If you are an administrator with the privilege of publishing resources, you are able to see the option Web Report Template from the File Type drop-down list. By selecting the option you can save the web report template in the current web report, the dialog appears



Template box

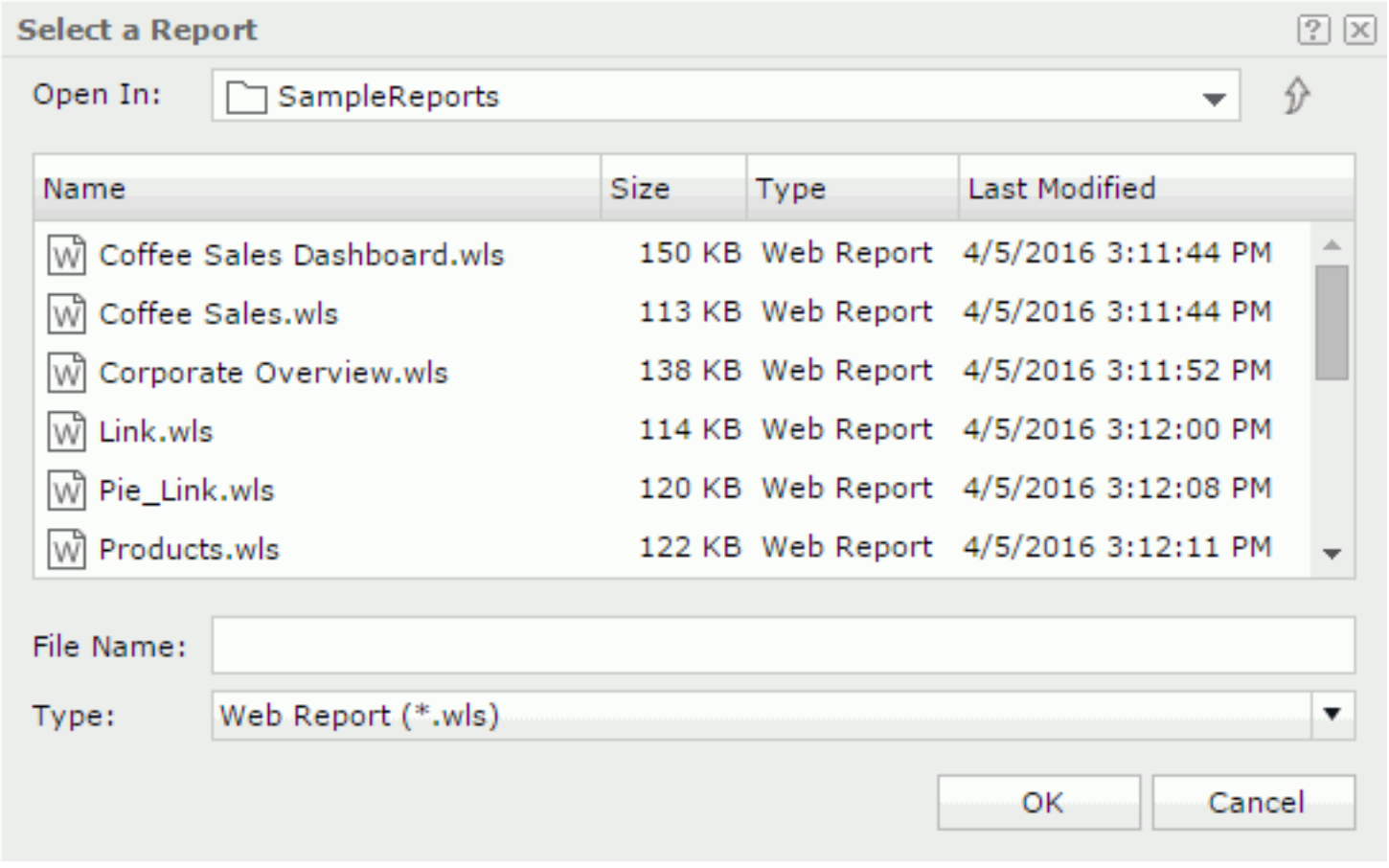
Lists the existing web report templates.

File Name

Specifies the name of the web report template. You can type the name as required to save it as a new wsld file or select one from the template box above to overwrite an existing wsld file according to Publish privilege, which is the system level authorization.

Select a Report dialog

This dialog appears when you click Menu > File > Open (or the Open button on the Standard toolbar), or click the Browse button in the Insert Link dialog or Edit Link dialog. It helps you to select the report you want to open or specify the target report which you want to link with the specified object.



Open

Specifies the catalog file which contains the report you want to open. When the dialog is opened by clicking the Browse button in the Insert Link dialog or Edit Link dialog, this option only displays the file name of the catalog which is used by the target linked report.

- **Name**
  Displays the file names.
- **Size**
  Displays the file size.
- **Type**
  Displays the file type.
- **Last Modified**
  Displays the last modified time of the file.

File Name

Specifies the name for the new file.

File Type

Specifies the type of the file.

OK

Applies the changes and closes the dialog.

Cancel

Does not retain any changes and closes the dialog.

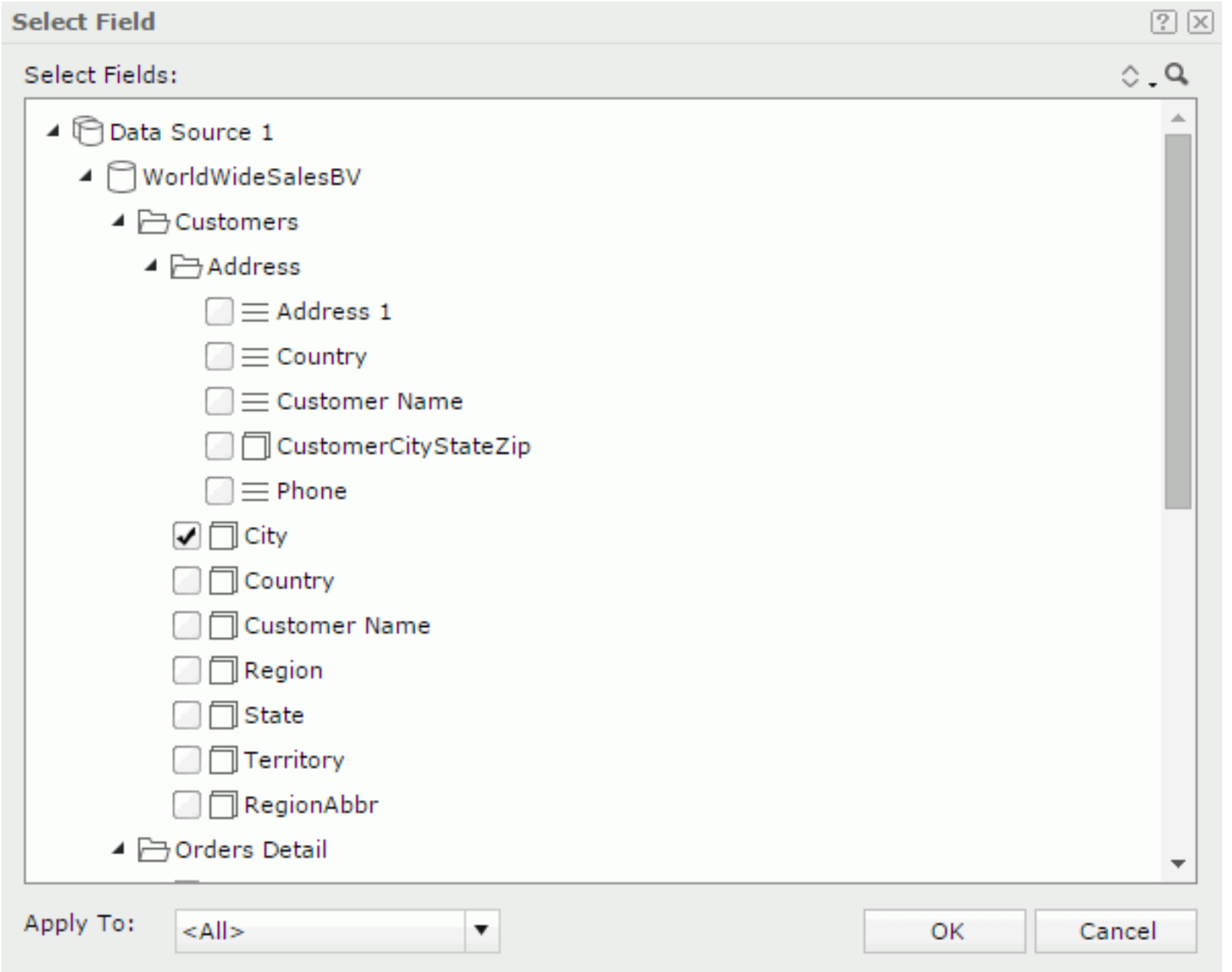
Displays the help document about this feature.

Ignores the setting and closes this dialog.

Select Field dialog

This dialog helps you to select the fields you want and varies according to different sources it is opened from.

When you click **+** on the title bar of the Filter panel to open the dialog, the options in the dialog are as follows.



Select Fields

Specifies the fields of the same type to add into the Filter panel, or to replace the current fields in the filter control. All uncomparable data type fields cannot be bound to filter controls, such as Binary, Blob, Clob, Longvarchar, Longvarbinary, and Varbinary.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.
- Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
- Highlights the previous matched text.
- Highlights the next matched text.

Apply To

Specifies the components to which the filter will be applied.

OK

Applies the selected field and closes the dialog.

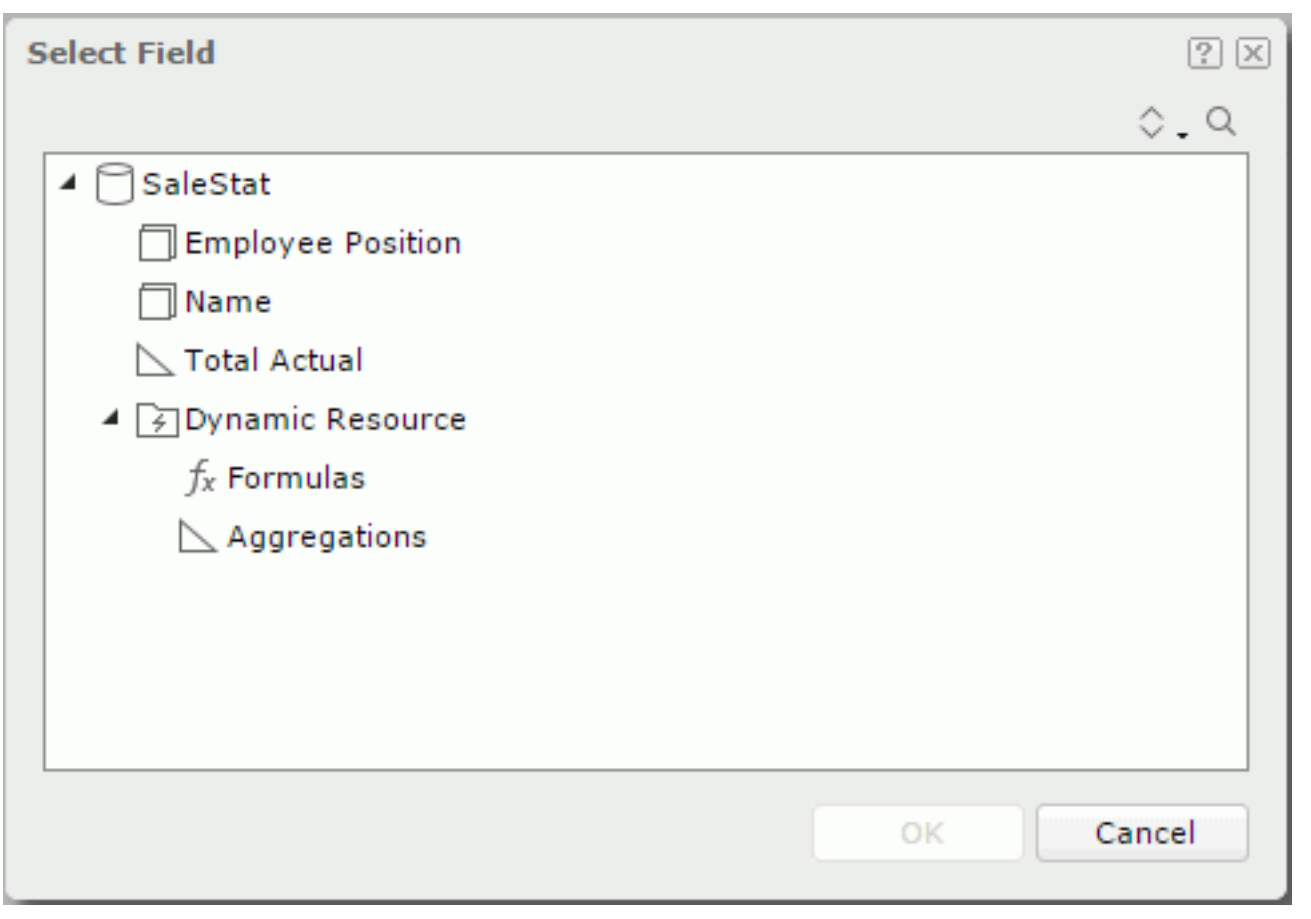
Cancel

Cancels the selection and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

When you click the Add Dynamic Field button in the Insert Link dialog or Edit Link dialog when the link type is URL or E-mail, the options in the dialog are as follows.



Resources box

Lists all the available data resources in the currently used dataset. Select a field from the resource tree to insert it into the URL or e-mail address to compose a dynamic hyperlink. You should make sure the field you select contains the information to compose a correct URL or e-mail address.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

Launches the quick search toolbar to search for view elements.

OK

Applies the selected field and closes the dialog.

Cancel

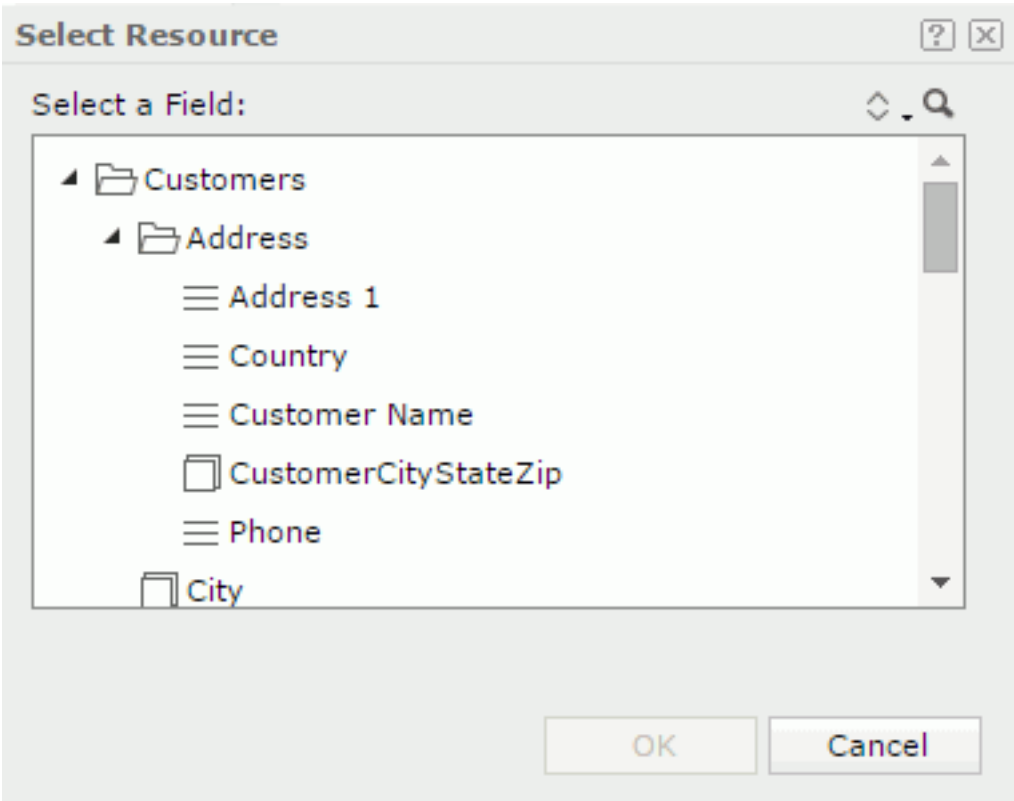
Cancels the selection and exits the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Select Resource dialog

This dialog appears when you click ... in the Add Aggregation dialog. It helps you to specify the field on which the dynamic aggregation will be based.



In this dialog, all the group objects and detail objects in the current business view and the dynamic formulas that have been created in the report will be listed. Select the required field and then click **OK**.

Note that a dynamic formula that is used as an aggregation cannot be applied as the mapping field of a dynamic aggregation.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.
- Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
- Highlights the previous matched text.
- Highlights the next matched text.

OK

Selects the field and closes this dialog.

Cancel

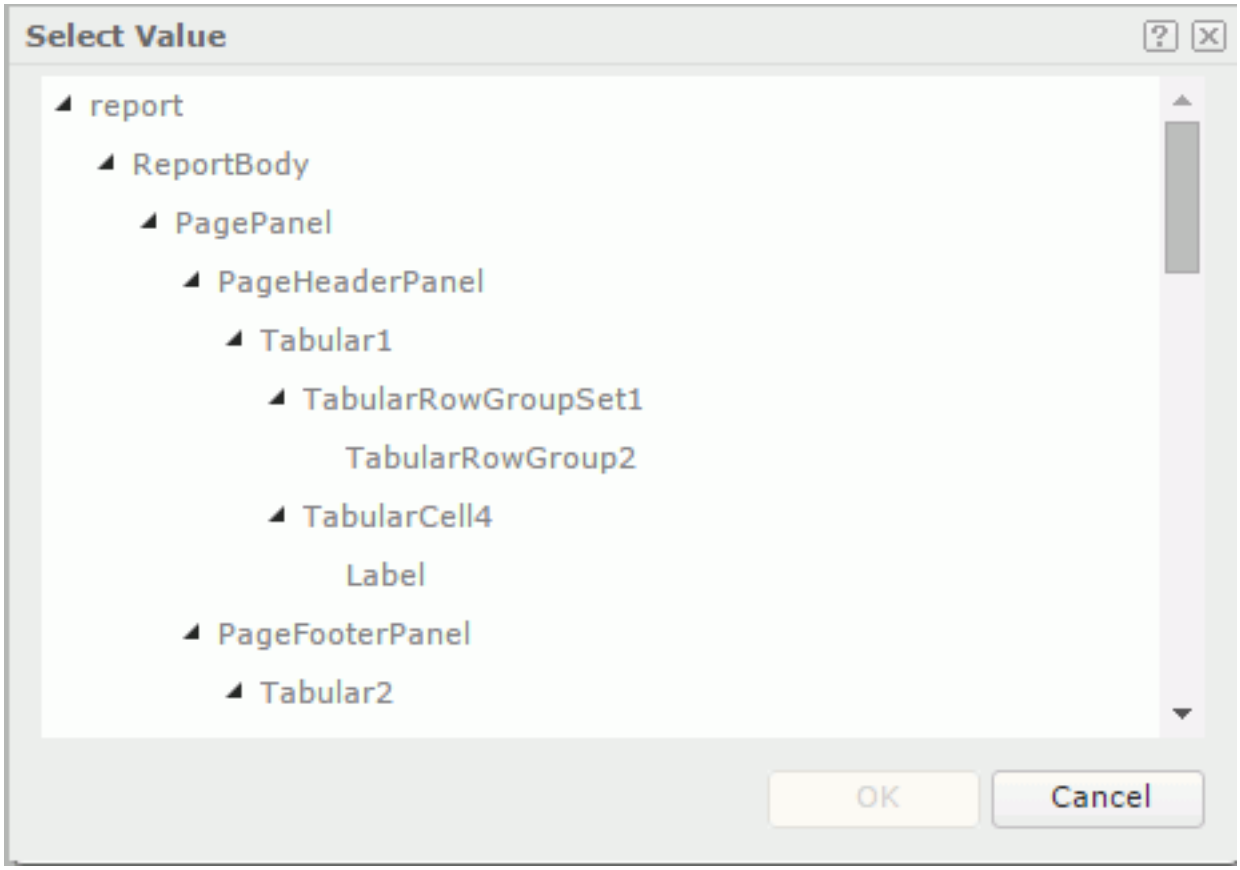
Cancels the selection of a field and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Select Value dialog

The dialog is displayed when you click the button in the Advanced tab for the Report link type in the Insert Link dialog or Edit Link dialog. It helps you to select a filter object such as filter control or slider to pass values between the filter objects in the primary report and the linked report.



The resources panel lists all the objects in the report. Only filter objects are available for selection. Select one to use.

OK

Selects a filter object and closes the dialog.

Cancel

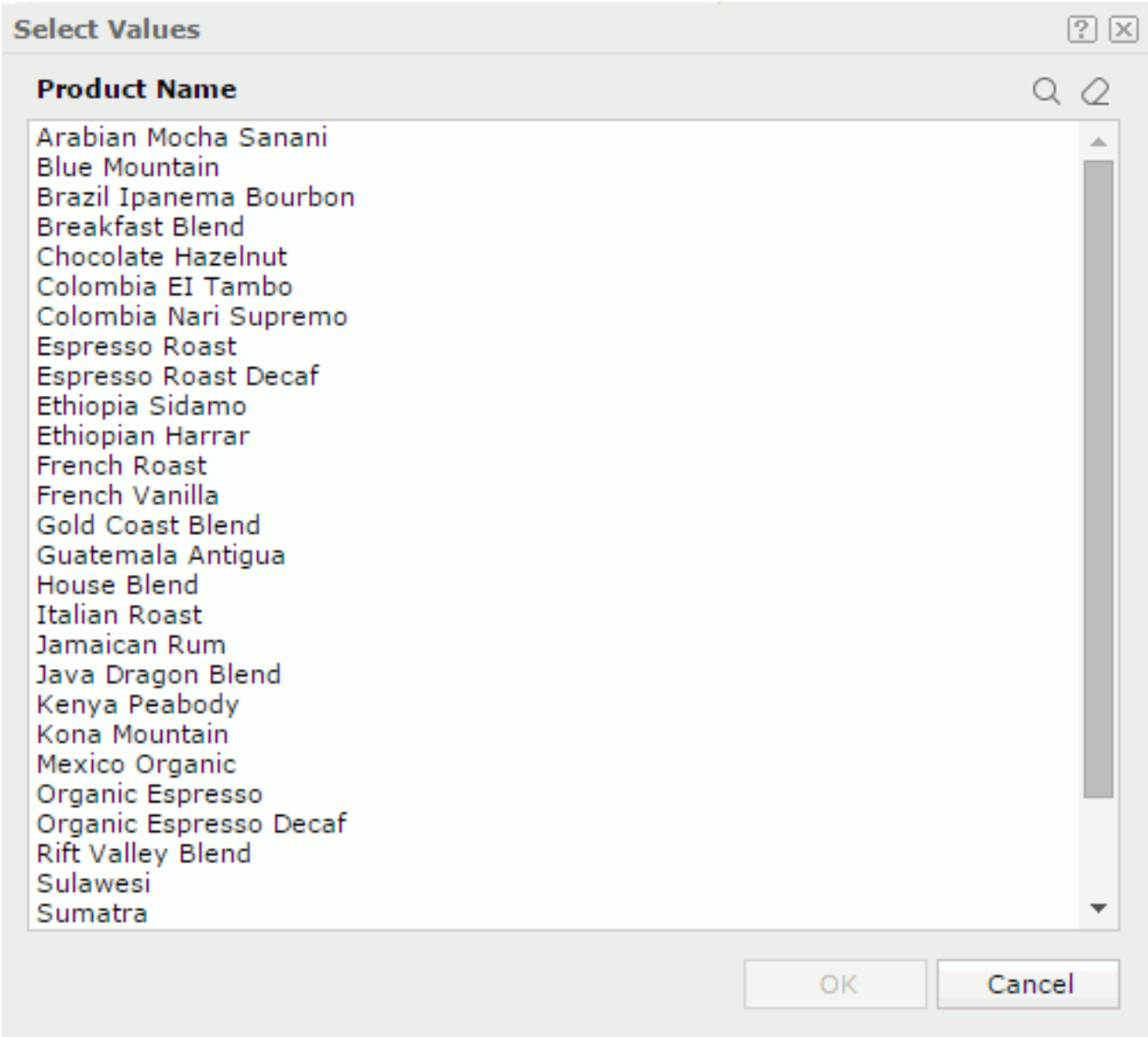
Cancels the selection and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Select Values dialog

This dialog appears when you right-click on any value of a table detail field and select Filter > More from the shortcut menu. It helps you to specify a value to apply a filter criterion.



Available Values

Specifies the values to filter data with. Multiple values can be selected.

When there are more than 300 values, JReport will use Big Data Loading logic. In this case, the Shift Key for multiple selection does not work.

Search

Launches the quick search toolbar to search for values.

- **Text box**
  The quick search toolbar treats field values as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the values.
- **X**
  Closes the quick search toolbar.
-
  Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
  - **Advanced**
    Displays the quick search toolbar in the advanced mode. With the advanced options, you can customize the value range as you want.

  The advanced options available on the search toolbar vary with field types:

  - **For fields of the String type**

    There are two search conditions. Select one and specify characters in the text box, then click the **OK** button on the search toolbar to start searching. To clear the characters you enter, click **X** in the text box. The Cancel button is used to close the search toolbar.

    - **Start with**
      Searches for the first characters among the values.
    - **End with**
      Searches for the last characters among the values.
  - **For fields of Numeric type**

    Five operators are supported for composing the search condition. Select one from the operator drop-down list and input value in the text box to compose the condition, then click the **OK** button on the search toolbar to start searching. To clear the value you enter, click **X** in the text box. The Cancel button is used to close the search toolbar.

    - **>**
      Greater than
    - **>=**
      Greater than or equal to
    - **<**
      Less than
    - **<=**
      Less than or equal to
    - **between**
      The data values are located between a range of values indicated by a start value and an end value. If selected, the second input value should not be smaller than the first input value, otherwise the search result will be null.
  - **For fields of Date/Time type**

    Five operators are supported for composing the search condition. Select one from the operator drop-down list and click the calendar button to select a value to compose the condition, then click the **OK** button on the search toolbar to start searching. The Cancel button is used to close the search toolbar.
-
  Highlights the previous matched text.
-
  Highlights the next matched text.

Cancels the selection of values.

OK

Closes the dialog and filters the field with the value you specified.

Cancel

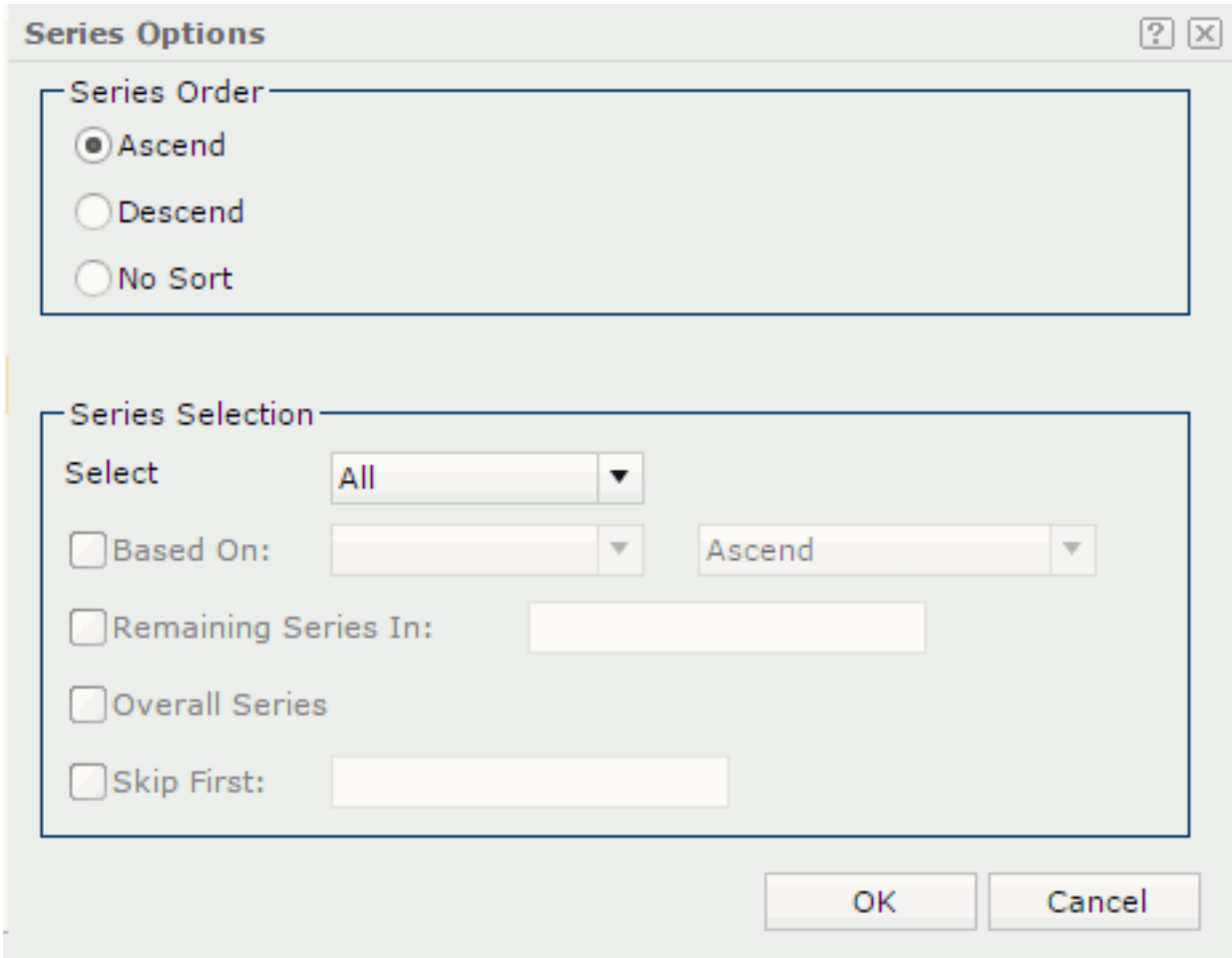
Cancels filtering the field and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Series Options dialog

This dialog appears when you click the Top N button above the Series box in the Bind Data screen of chart in the Web Report Wizard, Insert Chart dialog, Chart Wizard, or To Chart dialog. It helps you to set the sort order of the series values and define the number of the series values that will be displayed in the chart.



Series Order

Specifies in which order data on the series axis will be displayed.

- **Ascend**
  Lists data in an ascending order.
- **Descend**
  Lists data in a descending order.
- **No Sort**
  Keeps the data in their original order in database.

Series Selection

Specifies the number of the series values that will be displayed in the chart.

- **Select**
  Specifies the Select N condition to define the number of series values that will be displayed.
  - **All**
    If selected, all series values will be displayed.
  - **Top N**
    If selected, specify a number in the field to the right and the first N series values will be displayed.
  - **Bottom N**
    If selected, specify a number in the field to the right and the last N series values will be displayed.
- **Based On**
  If checked, the series values will be sorted by values of the summary selected with the direction specified. If unchecked, the series values will be sorted by the order specified in the Series Order box of the dialog.
- **Remaining Series In**
  Enabled when Top N or Bottom N is selected from the Select drop-down list. Check this option and then type a character string in the text field to group all the series values beyond the top/bottom N range.
- **Overall Series**
  Not supported on series values.
- **Skip First**
  If you check the Skip First option and input a number M in the text field to the right, then the first M series values in the chart will be skipped and the Select N condition will take effect beginning with M +1. The skipped values will be included in the Remaining Series group together with all the series values beyond the top/bottom N range.

OK

Accepts the changes and closes the dialog.

Cancel

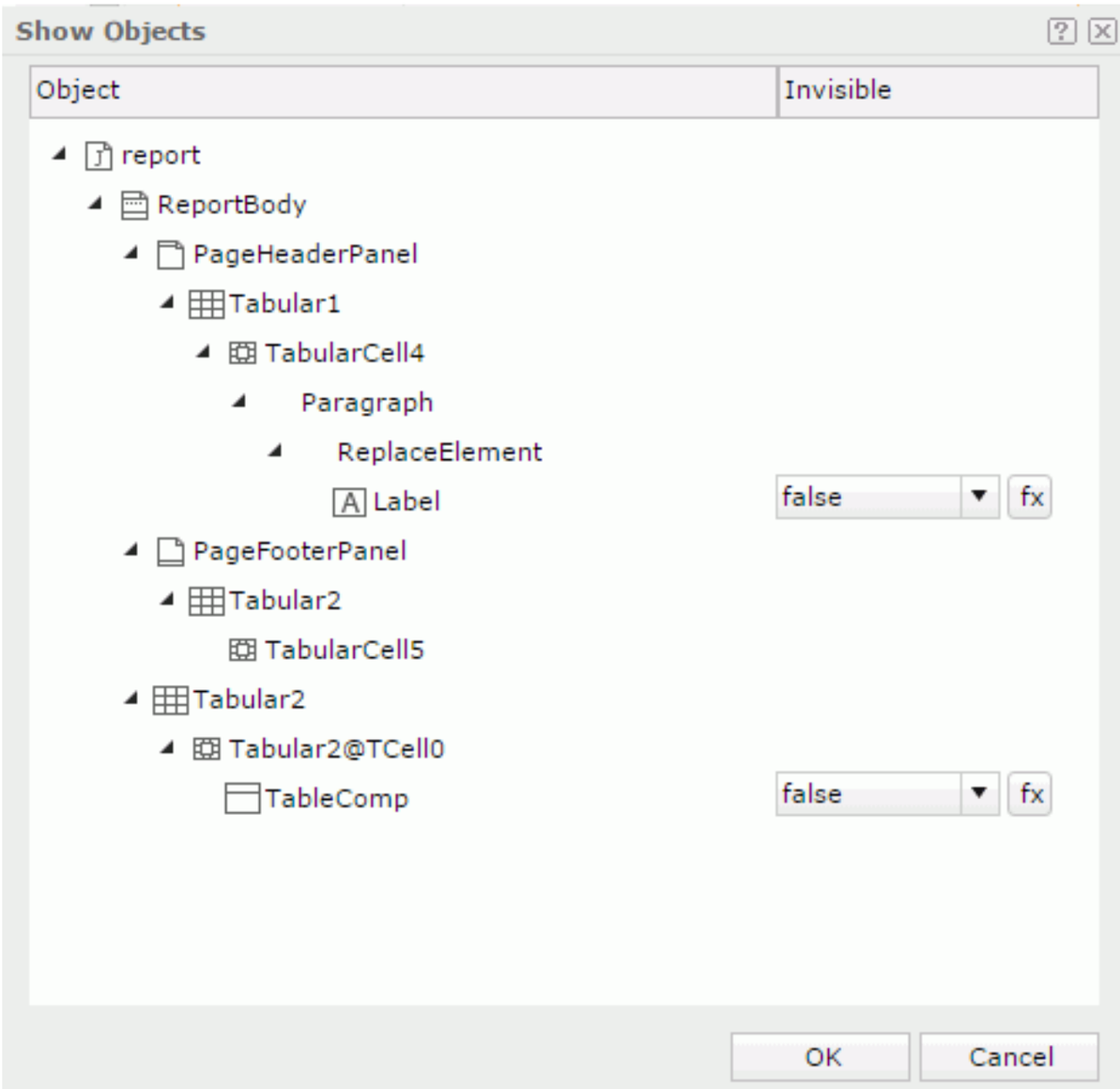
Discards the changes and closes the dialog.

Displays the help document about this feature.

Ignores the selection and closes this dialog.

Show Objects dialog

This dialog appears when you click the Show Objects button on the toolbar. It allows you specify which objects to show or hide in the web report.



Object

Lists the objects whose parents do not have data source in the report, for example, labels directly inserted into the tabular cells of the web report.

Invisible

Specifies whether to show or hide the objects. Select true or false from the drop-down list, or use a formula to control it.

OK

Accepts the changes and closes the dialog.

Cancel

Discards the changes and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Related topics:

- Using formulas to control showing or hiding components

Special Field Properties dialog

The dialog appears when you right-click a special field and select Properties from the shortcut menu. It helps you to edit the properties of the special field and contains the following tabs:

- General tab
- Font tab
- Border tab

OK

Applies the settings and closes this dialog.

Cancel

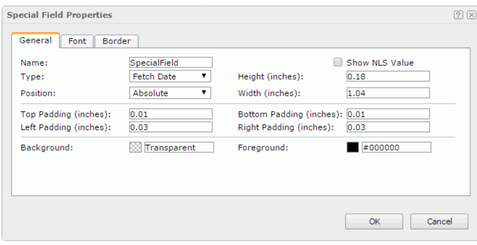
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the special field.



Name

Specifies the display name of the special field, which will be shown on the shortcut menu of the special field.

Show NLS Value

Specifies to show the translated name of the display name of the special field in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the special field is not modified.

Width

Specifies the width of the special field.

Type

Specifies the type of the special field. Select a new one from the drop-down list if you want to change the type. The following types are available:

- **User Name**
  Prints the user name with which you log onto JReport Server.
- **Modified Date**
  Prints the last modified date for the catalog.
- **Modified Time**
  Prints the last modified time for the catalog.
- **Print Date**
  Prints today's date (or the current date from your computer).
- **Print Time**
  Prints the current time from your computer.
- **Fetch Date**
  Prints the date when the data is retrieved from the database.
- **Fetch Time**
  Prints the time when the data is retrieved from the database.

Height

Specifies the height of the special field.

Position

Specifies the position mode of the special field. If the special field is directly contained in the report body, a tabular cell, or a text box, its position mode can be modified.

- **Absolute**: The special field's position will be decided by its X and Y property values.
- **Static**: The special field will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled.

Top Padding

Specifies the space between the text of the special field and its top border.

Bottom Padding

Specifies the space between the text of the special field and its bottom border.

Left Padding

Specifies the space between the text of the special field and its left border.

Right Padding

Specifies the space between the text of the special field and its right border.

Background

Specifies the background color of the special field.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

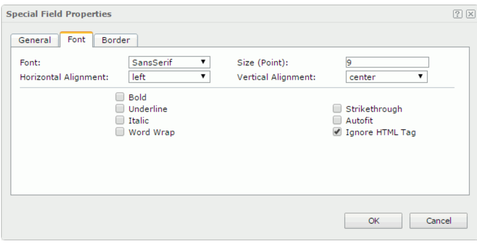
Foreground

Specifies the foreground color of the special field.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB.

Font tab

This tab shows the font-related information of the special field. You can modify all the font settings in this tab.



Font

Specifies the font face of the special field text.

Size

Specifies the font size of the special field text.

Horizontal Alignment

Specifies the horizontal alignment mode of the text in the special field.

Vertical Alignment

Specifies the vertical alignment mode of the text in the special field.

Bold

Specifies whether to make the object text bold or not.

Underline

Specifies whether the field text will be underlined or not.

Strikethrough

Specifies whether or not to attach a strikeout line to the field text.

Italic

Specifies whether to make the field text italic or not.

Autofit

Specifies whether or not to automatically adjust the width of the special field.

Word Wrap

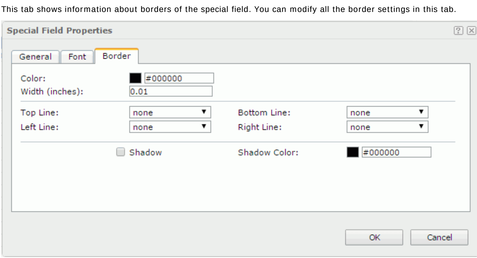
Specifies whether or not to wrap the text to the special field width.

Ignore HTML Tag

If this option is unchecked, JReport will parse HTML tag elements in the field value while the report is to be saved as an HTML file; or the field value will appear in the HTML file the same as that in Web Report Studio (HTML tag elements in the field value, if any, will not be parsed).

Border tab

This tab shows information about borders of the special field. You can modify all the border settings in this tab.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Shadow

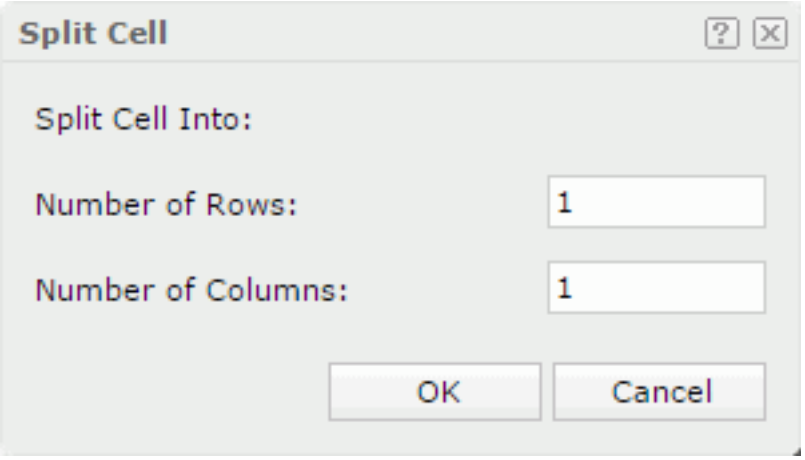
Specifies whether the borders will have a shadow effect or not.

Shadow Color

Specifies the color of the border shadow.

Split Cell dialog

The dialog appears when you select a tabular cell and click Menu > Format > Split. It helps you to specify the number of rows and columns that the specified tabular cell will be split into.



Split Cell Into

Specifies the number of rows and columns to split the selected tabular cell into.

- **Number of Rows**
  Specifies the number of rows the tabular cell will be split into. Input a positive integer in the text box to specify the number.
- **Number of Columns**
  Specifies the number of columns the tabular cell will be split into. Input a positive integer in the text box to specify the number.

OK

Closes this dialog and splits the tabular cell.

Cancel

Cancels to split the cell and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Summary Properties dialog

The dialog appears when you right-click a summary field and select Properties from the shortcut menu. It helps you to specify the properties of the summary field and contains the following tabs:

- General tab
- Font tab
- Border tab

OK

Applies the settings and closes this dialog.

Cancel

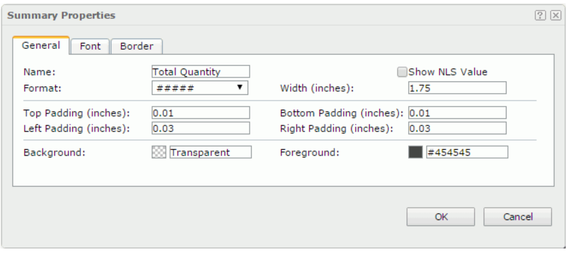
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the summary field.



Name

Specifies the display name of the summary field.

Show NLS Value

Specifies to show the translated name of the display name of the summary field in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the summary field is not modified.

Width

Specifies the width of the summary field.

Format

Specifies the format of the summary field.

Height

Specifies the height of the summary field.

Top Padding

Specifies the space between the text of the summary field and its top border.

Bottom Padding

Specifies the space between the text of the summary field and its bottom border.

Left Padding

Specifies the space between the text of the summary field and its left border.

Right Padding

Specifies the space between the text of the summary field and its right border.

Background

Specifies the background color of the summary field.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

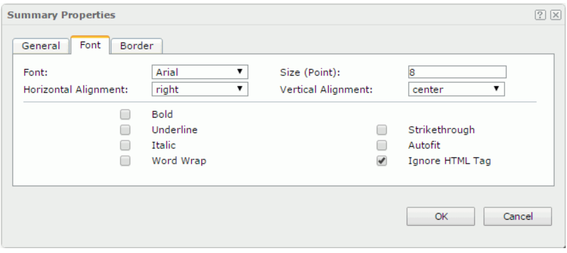
Foreground

Specifies the foreground color of the summary field.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB.

Font tab

This tab shows the font-related information of the summary field.



Font

Specifies the font face of the text.

Size

Specifies the font size of the text.

Horizontal Alignment

Specifies the horizontal alignment mode of the text in the summary field.

Vertical Alignment

Specifies the vertical alignment mode of the text in the summary field.

Bold

Specifies whether to make the text bold or not.

Underline

Specifies whether the text will be underlined or not.

Strikethrough

Specifies whether or not to attach a strikeout line to the text.

Italic

Specifies whether to make the text italic or not.

Autofit

Specifies whether or not to automatically adjust the width of the summary field.

Word Wrap

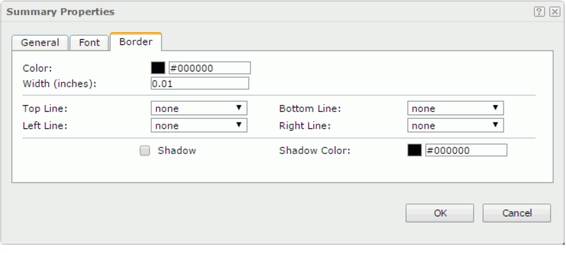
Specifies whether or not to wrap the text to the summary field width.

Ignore HTML Tag

If this option is unchecked, JReport will parse HTML tag elements in the field value while the report is to be saved as an HTML file; or the field value will appear in the HTML file the same as that in Web Report Studio (HTML tag elements in the field value, if any, will not be parsed).

Border tab

This tab shows information about borders of the summary field.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Shadow

Specifies whether the borders will have a shadow effect or not.

Shadow Color

Specifies the color of the border shadow.

Table Cell Properties dialog

The dialog appears when you right-click a table cell and select Properties from the shortcut menu. It helps you to specify the properties of the table cell and contains the following tabs:

- General tab
- Border tab

OK

Applies the settings and closes this dialog.

Cancel

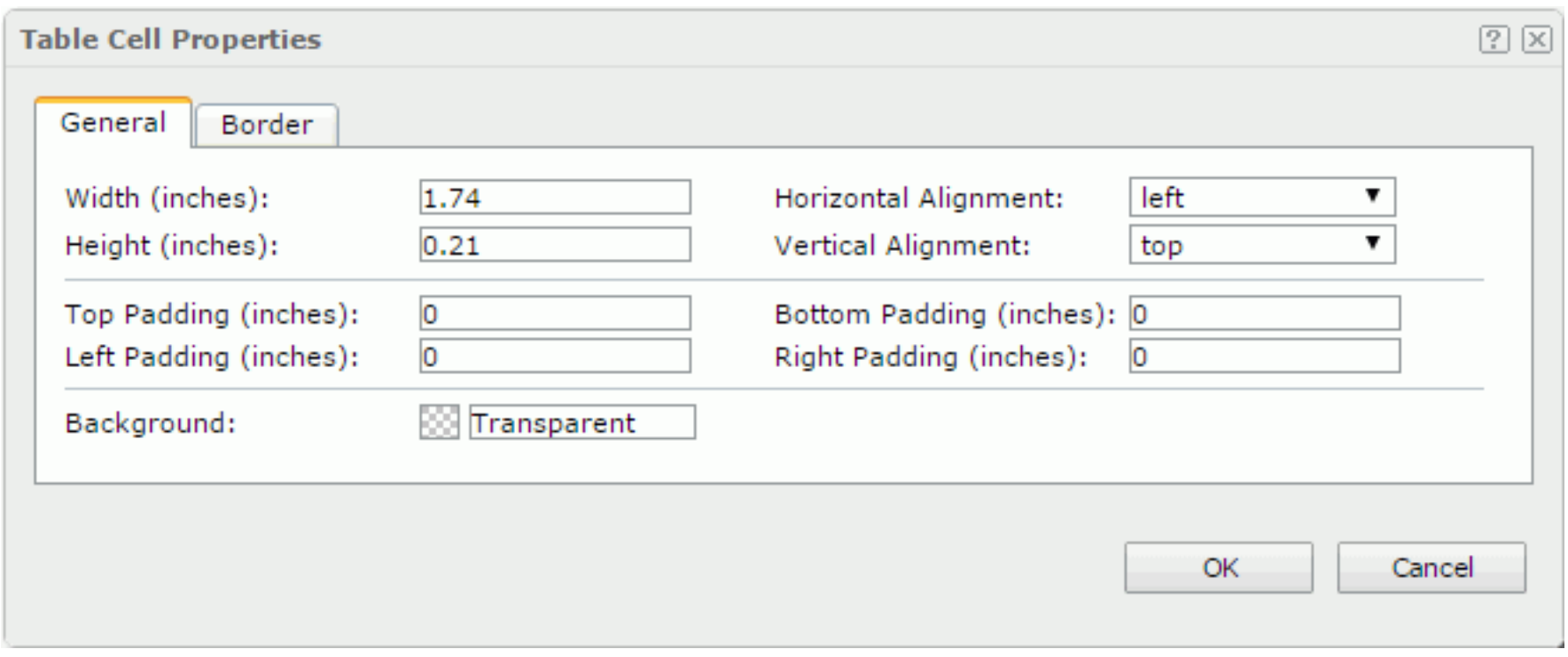
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the table cell.



Width

Specifies the width of the table cell.

Horizontal Alignment

Specifies the horizontal alignment mode of the text in the table cell.

Height

Specifies the height of the table cell.

Vertical Alignment

Specifies the vertical alignment mode of the text in the table cell.

Top Padding

Specifies the space between the text of the table cell and its top border.

Bottom Padding

Specifies the space between the text of the table cell and its bottom border.

Left Padding

Specifies the space between the text of the table cell and its left border.

Right Padding

Specifies the space between the text of the table cell and its right border.

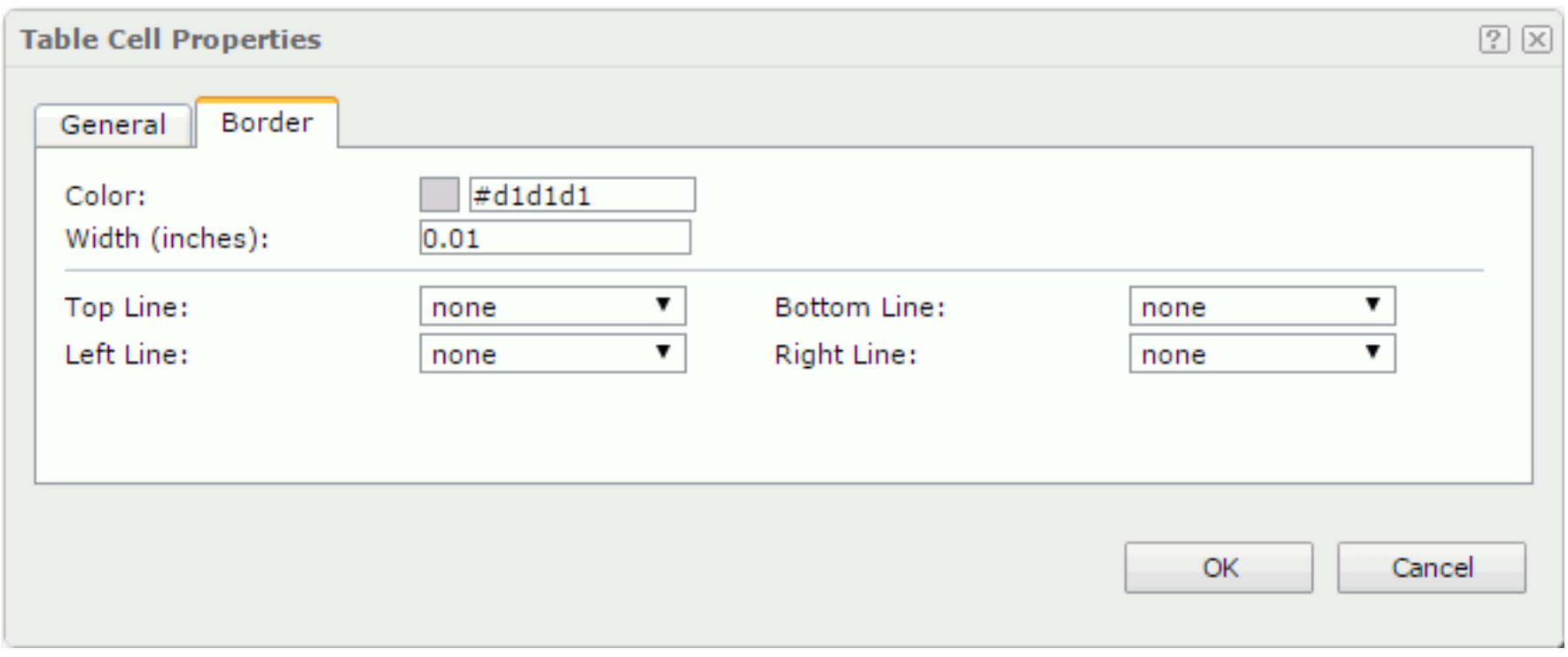
Background

Specifies the background color of the table cell.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the table cell.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Table Detail Properties dialog

The dialog appears when you click anywhere in a table detail row, then click the button that appears on the table detail to select the row, right-click on the row and select Properties from the shortcut menu. It helps you to specify the properties of the table detail row and contains the following tabs:

- General tab
- Border tab

OK

Applies the settings and closes this dialog.

Cancel

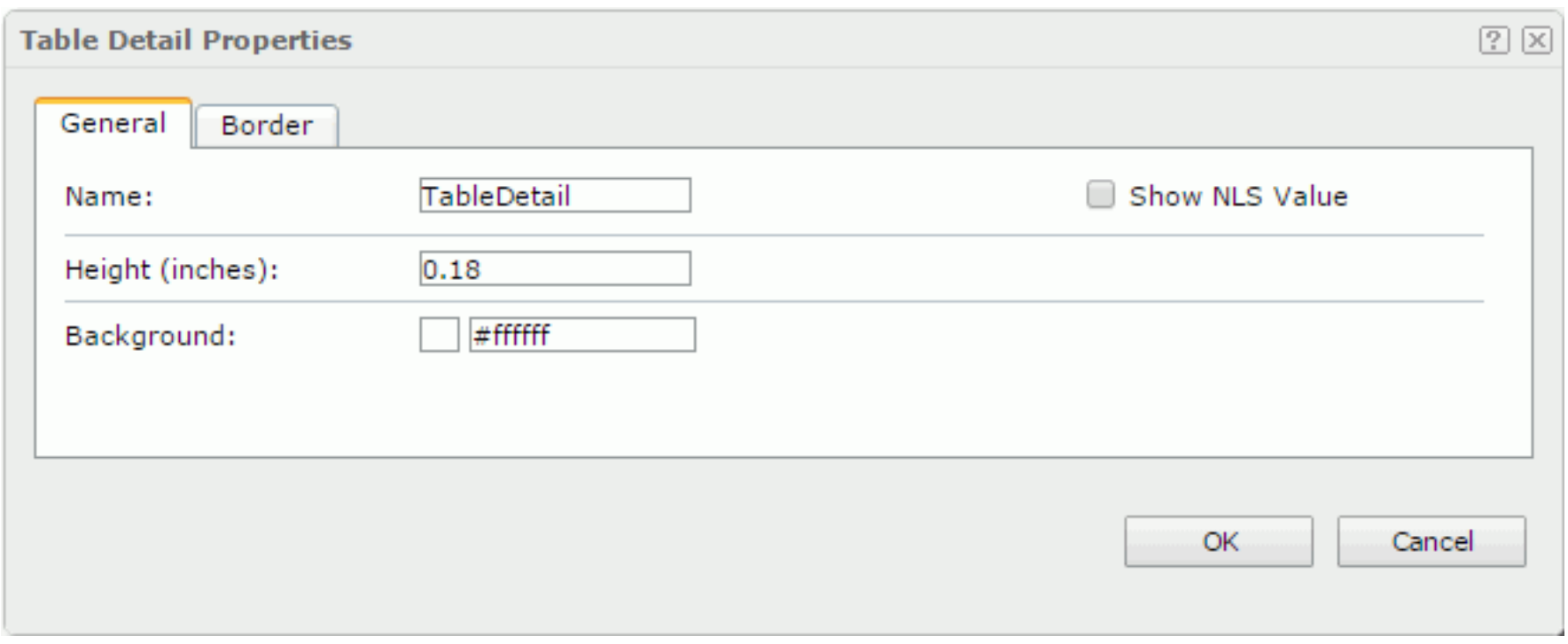
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the table detail row.



Name

Specifies the display name of the table detail row.

Show NLS Value

Specifies to show the translated name of the display name of the table detail row in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the table detail row is not modified.

Height

Specifies the height of the table detail row.

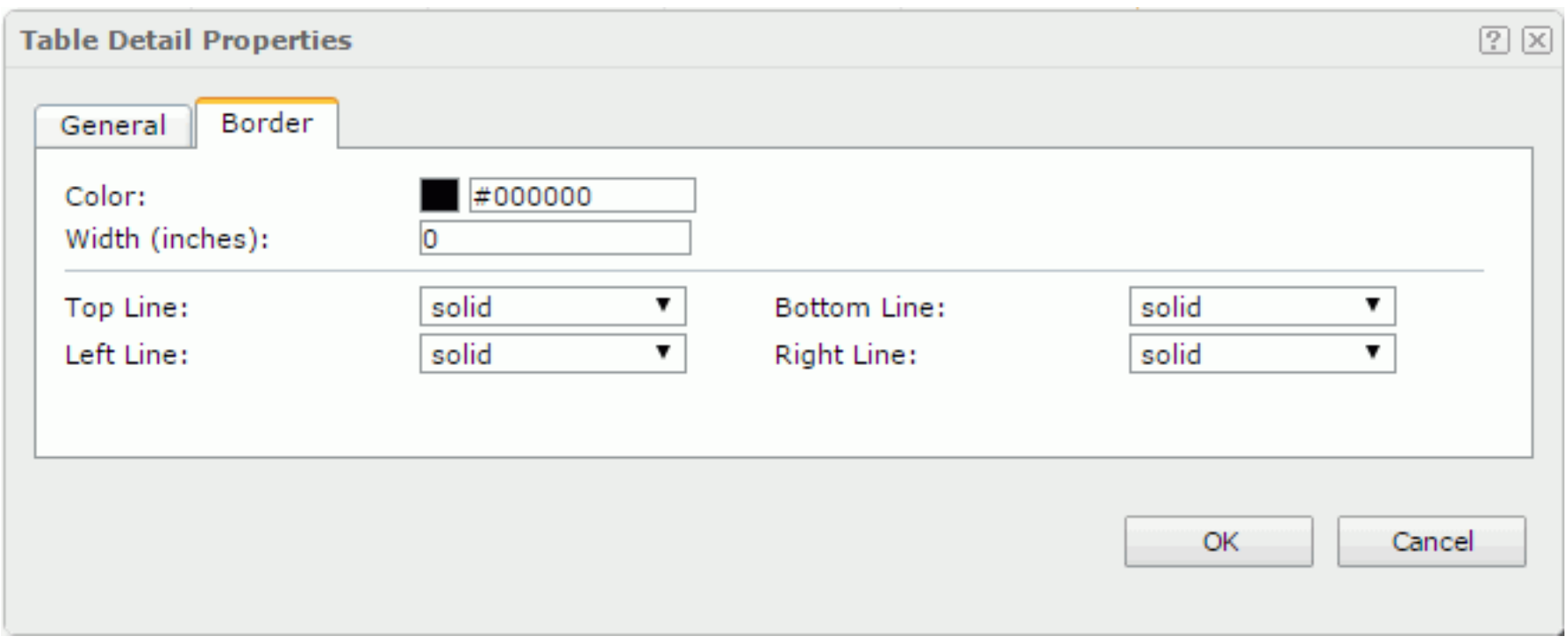
Background

Specifies the background color of the table detail row.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB, or click **More Fill Effects** to specify a gradient or an image as the fill effect in the Fill Effects dialog. If you want to make the color transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the table detail row.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Table Footer Properties dialog

The dialog appears when you click anywhere in a table footer row, then click the button that appears on the table footer to select the row, right-click on the row and select Properties from the shortcut menu. It helps you to specify the properties of the table footer and contains the following tabs:

- General tab
- Border tab

OK

Applies the settings and closes this dialog.

Cancel

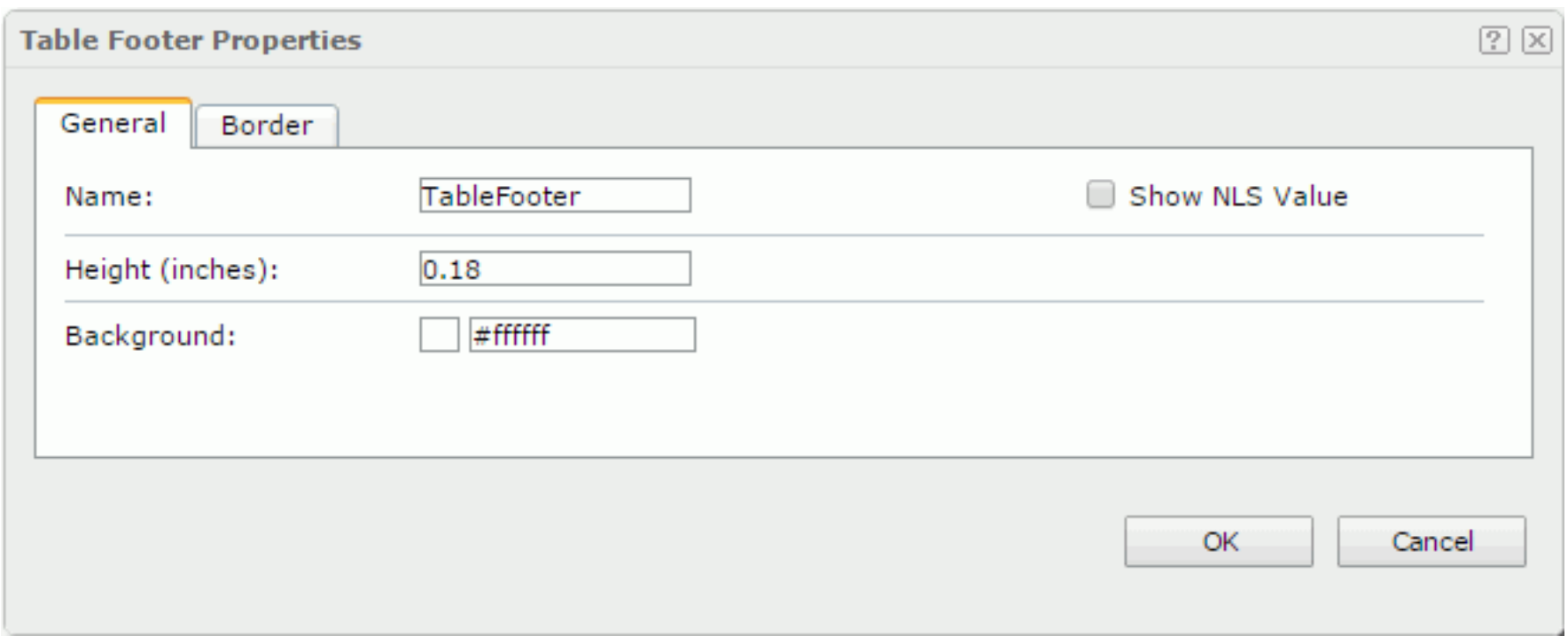
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the table footer.



Name

Specifies the display name of the table footer.

Show NLS Value

Specifies to show the translated name of the display name of the table footer in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the table footer is not modified.

Height

Specifies the height of the table footer.

Background

Specifies the background color of the table footer.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB, or click **More Fill Effects** to specify a gradient or an image as the fill effect in the Fill Effects dialog. If you want to make the background transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the table footer.

Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Table Header Properties dialog

The dialog appears when you click anywhere in a table header row, then click the button that appears on the table header to select the row, right-click on the row and select Properties from the shortcut menu. It helps you to specify the properties of the table header and contains the following tabs:

- General tab
- Border tab

OK

Applies the settings and closes this dialog.

Cancel

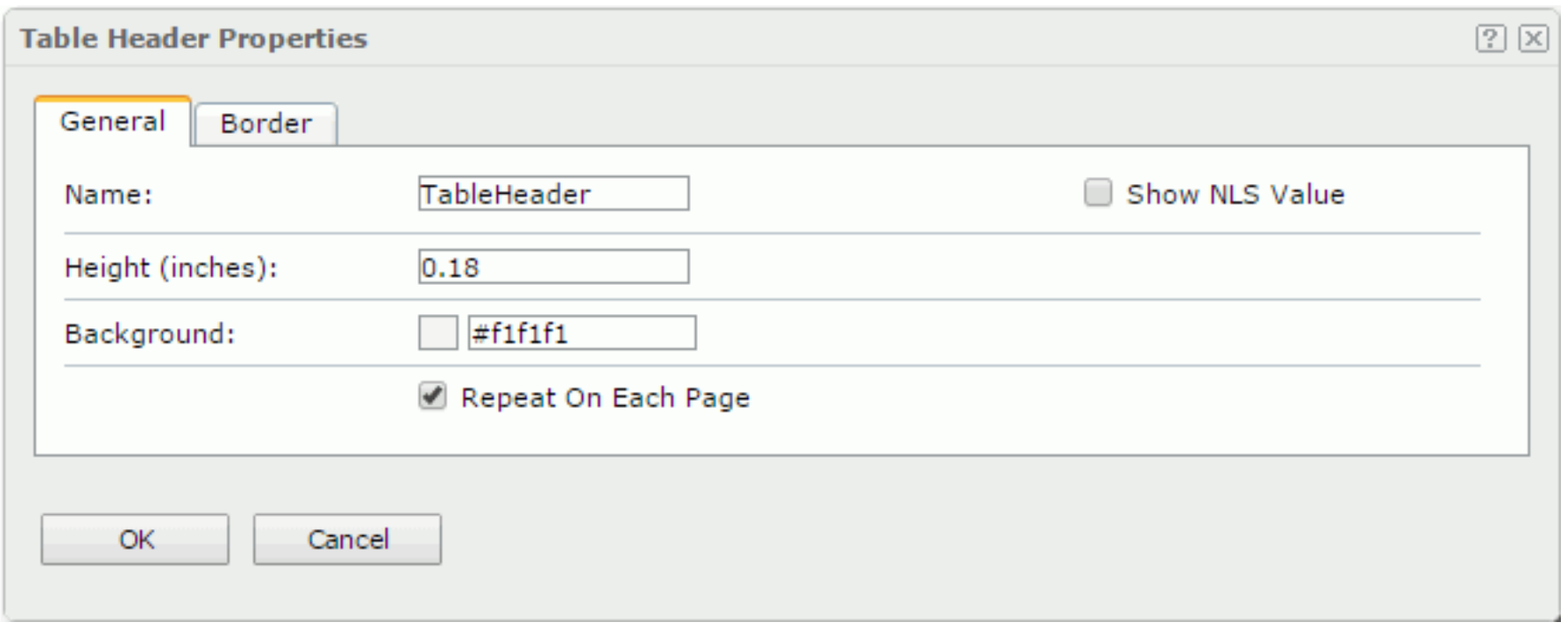
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the table header.



Name

Specifies the display name of the table header.

Show NLS Value

Specifies to show the translated name of the display name of the table header in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the table header is not modified.

Height

Specifies the height of the table header.

Background

Specifies the background color of the table header.

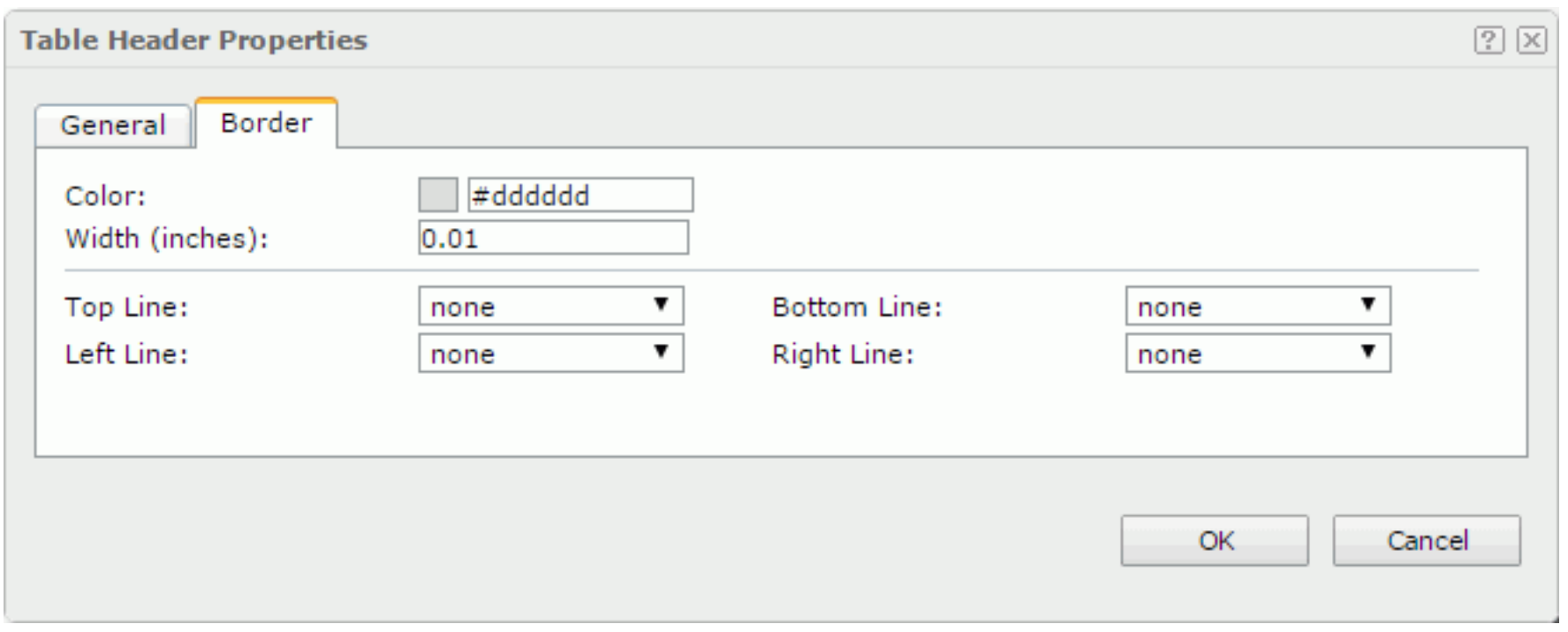
To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB, or click **More Fill Effects** to specify a gradient or an image as the fill effect in the Fill Effects dialog. If you want to make the background transparent, input Transparent in the text box.

Repeat On Each Page

Specifies whether to repeat the table header on each page of the report.

Border tab

This tab shows information about borders of the table header.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Table Properties dialog

The dialog appears when you right-click a table and select Properties from the shortcut menu. It helps you to specify the properties of the table and contains the following tabs:

- General tab
- Border tab

OK

Applies the settings and closes this dialog.

Cancel

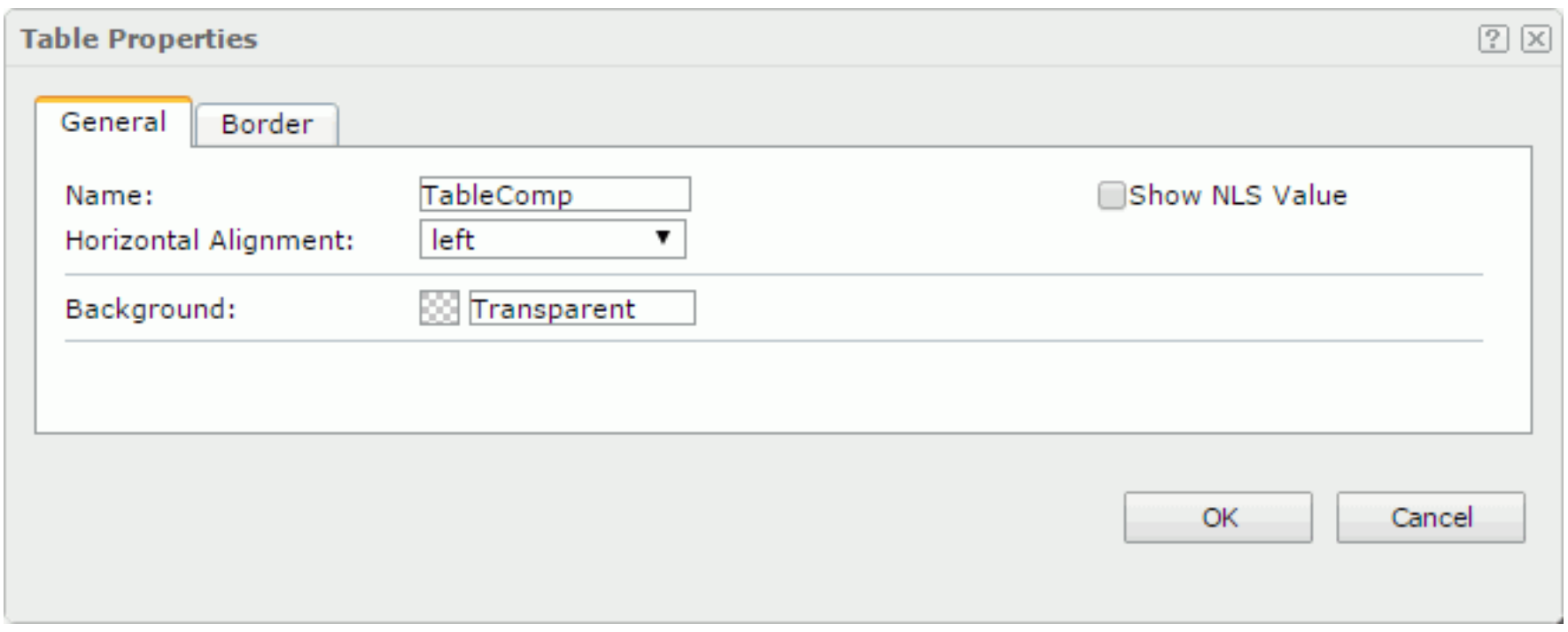
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the table.



Name

Specifies the display name of the table, which will be shown on the shortcut menu of the table.

Show NLS Value

Specifies to show the translated name of the display name of the table in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the table is not modified.

Horizontal Alignment

Specifies the horizontal justification of the table. Choose an option from the drop-down list.

- **left**: Aligns the table on the left of the report.
- **right**: Aligns the table on the right of the report.
- **center**: Aligns the table in the center of the report.

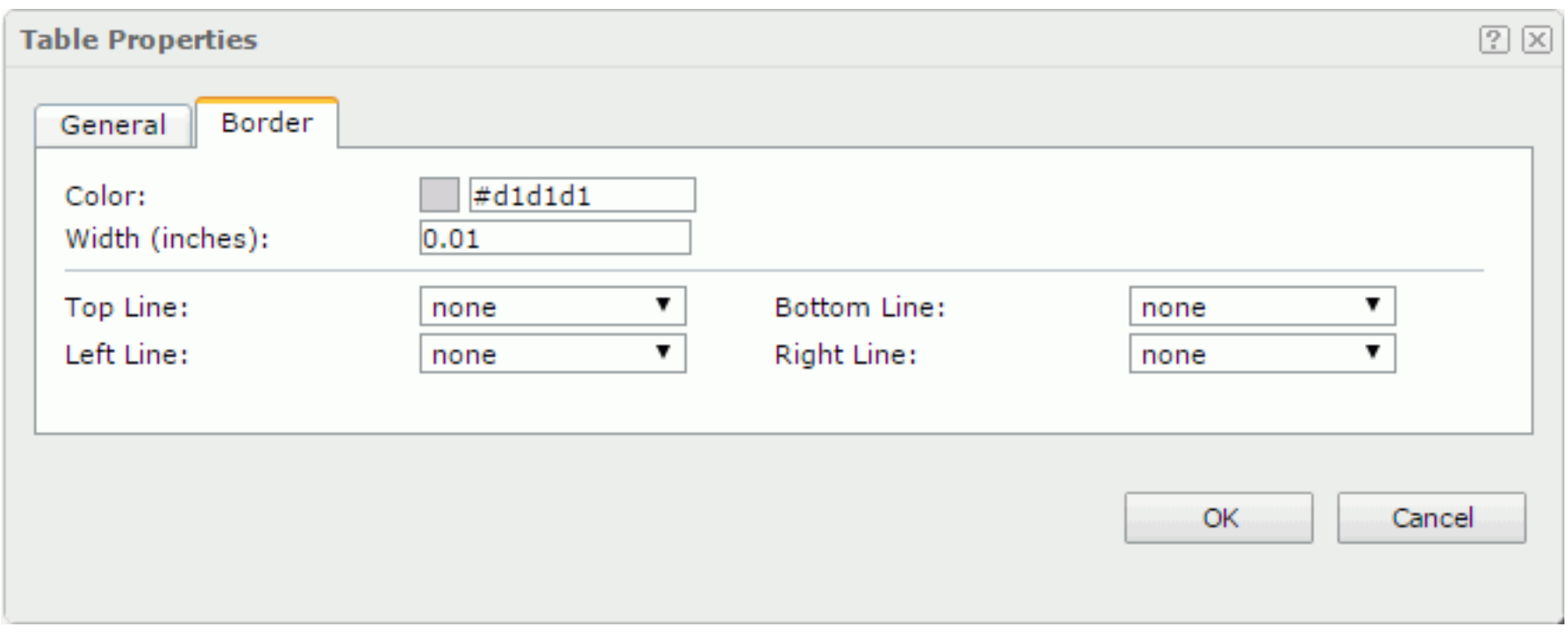
Background

Specifies the background color of the table.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

Border tab

This tab shows information about borders of the table.



Color

Specifies the border color.

Width

Specifies the border width.

Top Line

Specifies the style of the top border line.

Bottom Line

Specifies the style of the bottom border line.

Left Line

Specifies the style of the left border line.

Right Line

Specifies the style of the right border line.

Table Wizard

This wizard is displayed when you do one of the following:

- Select a table, then click Menu > Edit > Wizard.
- Select a table, then click the Table Wizard button on the Context toolbar.
- Right-click the icon of a table and select Table Wizard from the shortcut menu.

It helps you to change data of the table and consists of the following tabs:

- Display tab
- Group tab

Table Title

Specifies a title for the table.

Specifies the font properties of the table title.

- **Font**
  Lists all the available font faces that can be selected to apply to the title.
- **Font Style**
  Specifies the font style of the title. It can be one of the following: plain, bold, italic, and bold italic.
- **Size**
  Specifies the font size of the title.
- **Align**
  Specifies the position of the title to be left, right, center or justify.
- **Font Color**
  Specifies the font color of the title.
- **Background Color**
  Specifies the background color of the title.

Data Source

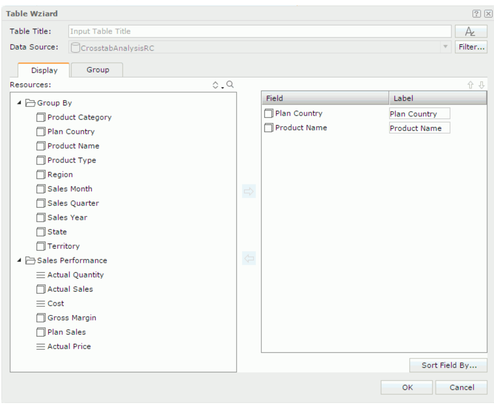
Displays the business view that has been used in the table.

Filter

Opens the Query Filter dialog to specify the filter which you want to apply to the selected business view.

Display tab

Specifies the fields to be displayed in the table.



Resources

Displays all the view elements in the selected business view.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.
- Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
- Highlights the previous matched text.
- Highlights the next matched text.

Adds the selected view element to be displayed in the table.

Removes the selected view element.

Field

Lists the view elements that have been added to the table.

Label

Specifies the display names of the selected fields.

Moves the selected view element one step up.

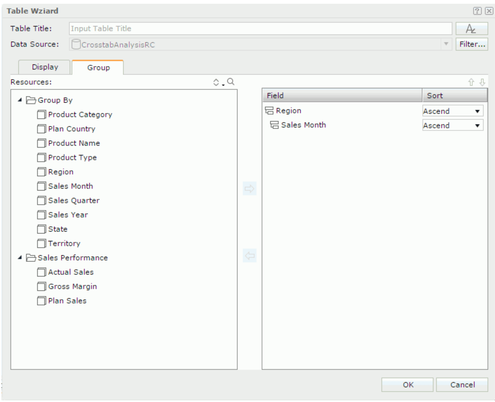
Moves the selected view element one step down.

Sort Fields By

Opens the Custom Sort dialog to specify how to sort data in the table.

Group tab

Specifies the fields to group the data.



Resources

Displays all the available group objects you can use to group the data in the table.

Sorts the group objects in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

Launches the quick search toolbar to search for group objects.

Adds the selected group object as a group field.

Removes the selected group object.

Field

Lists all the group objects that have been added as the group fields.

Sort

Specifies the sort order for groups at the specific group level.

- **Ascend**
  Groups will be sorted in an ascending order (A, B, C).
- **Descend**
  Groups will be sorted in a descending order (C, B, A).
- **No Sort**
  Groups will be sorted in the original order in database.
- **Custom Sort**
  Opens the Custom Sort dialog to set how groups will be sorted.

Moves the selected group one step up.

Moves the selected group one step down.

OK

Applies the changes and closes the wizard.

Cancel

Does not retain changes and closes this wizard.

Displays the help document about this feature.

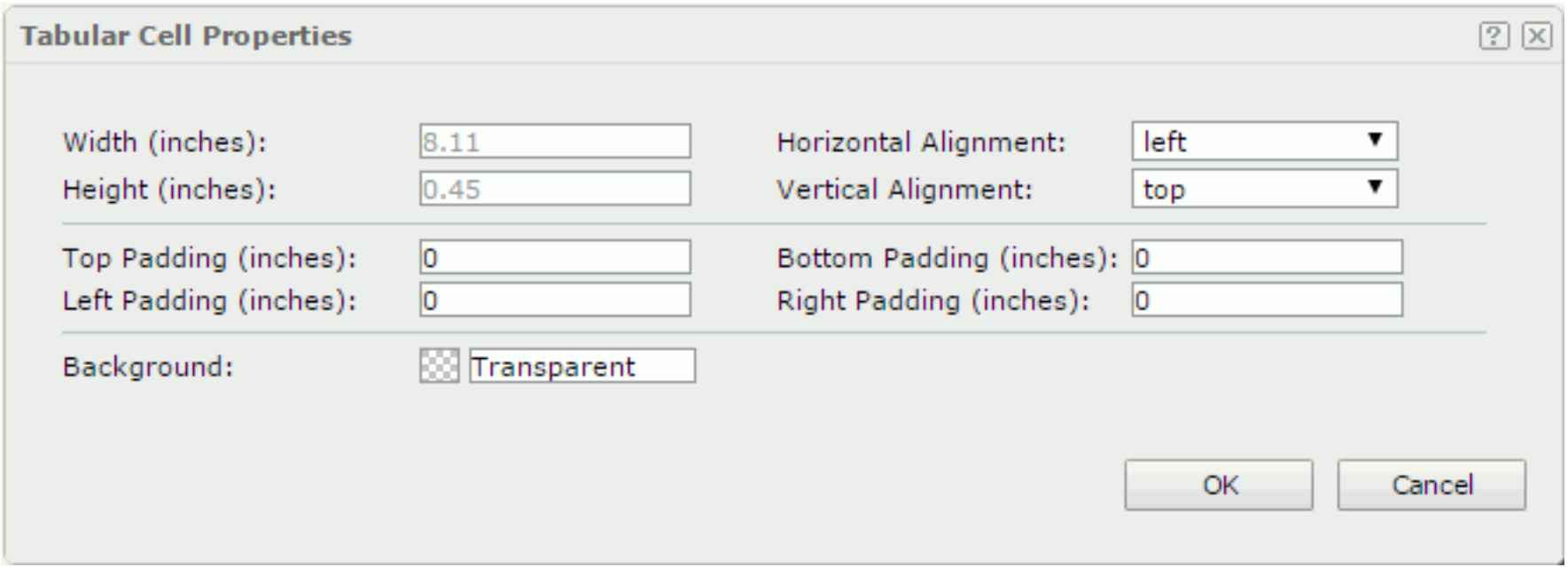
Ignores the setting and closes this dialog.

Related topics:

- Changing the table definition

Tabular Cell Properties dialog

The dialog appears when you right-click a tabular cell and select Properties from the shortcut menu. It helps you to specify the properties of the tabular cell.



Width

Specifies the width of the tabular cell.

Horizontal Alignment

Specifies the horizontal alignment mode of the text in the tabular cell.

Height

Specifies the height of the tabular cell.

Vertical Alignment

Specifies the vertical alignment mode of the text in the tabular cell.

Top Padding

Specifies the space between the text of the tabular cell and its top border.

Bottom Padding

Specifies the space between the text of the tabular cell and its bottom border.

Left Padding

Specifies the space between the text of the tabular cell and its left border.

Right Padding

Specifies the space between the text of the tabular cell and its right border.

Background

Specifies the background color of the tabular cell.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

OK

Applies the settings and closes this dialog.

Cancel

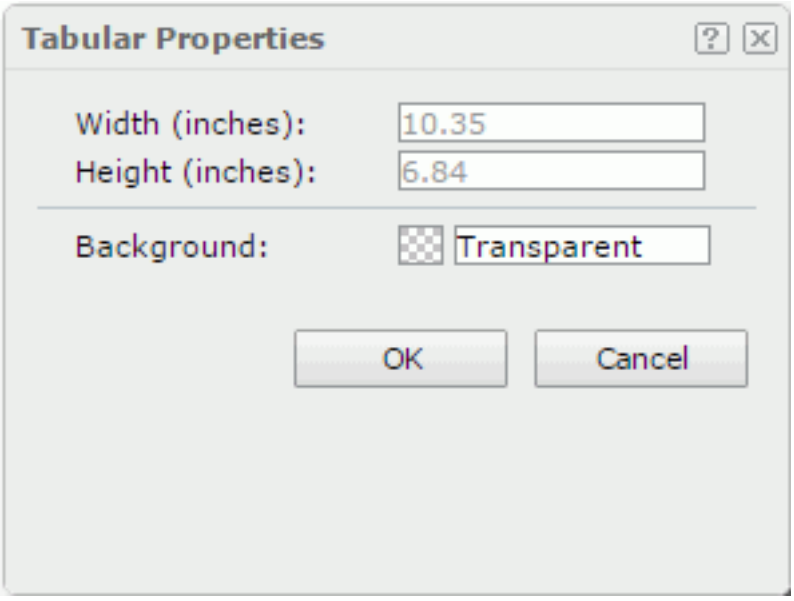
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Tabular Properties dialog

The dialog appears when you right-click a tabular and select Properties from the shortcut menu. It helps you to edit the properties of the tabular.



Width

Specifies the width of the tabular.

Height

Specifies the height of the tabular.

Background

Specifies the background color of the tabular.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB. If you want to make the background transparent, input Transparent in the text box.

OK

Applies the settings and closes this dialog.

Cancel

Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

To Chart dialog

This dialog appears when you right-click a crosstab and then select To Chart on the shortcut menu or click Menu > Edit > To Chart. It allows you to specify settings for converting a crosstab into a chart excluding the organization chart and varies with different chart types: common chart types, heat map.

OK

Applies the settings and closes the dialog.

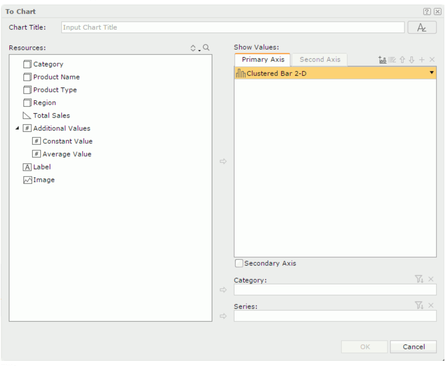
Cancel

Cancels the settings and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

For common chart types



Title

Specifies a title for the chart.

Specifies the font properties of the chart title.

- **Font**
  Lists all the available font faces that can be selected to apply to the title.
- **Font Style**
  Specifies the font style of the title. It can be one of the following: plain, bold, italic, and bold italic.
- **Size**
  Specifies the font size of the title.
- **Align**
  Specifies the position of the title to be left, right, center or justify.
- **Font Color**
  Specifies the font color of the title.
- **Background Color**
  Specifies the background color of the title.

Resources

Displays all the view elements used in the crosstab.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.
-
- Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
-
- Highlights the previous matched text.
-
- Highlights the next matched text.

Adds the selected group or aggregation object to be displayed in the chart.

Category

Lists the group object that will be displayed on the category axis of the chart.

Series

Lists the group object that will be displayed on the series axis of the chart.

Show Values

Specifies the types for chart and sets the values for the Primary Axis or Secondary Axis separately.

- **Primary Axis**
  Adds a chart type to the primary axis.
- **Secondary Axis**
  Adds a chart type to the secondary axis.
- **X Axis**
  Lists the value you want to show on the X axis of the bubble chart.
- **Y Axis**
  Lists the value you want to show on the Y axis of the bubble chart.
- **Radius**
  Lists the value you want to show as the bubble radius.

Secondary Axis

Specifies whether to show the secondary axis in the chart.

Opens the Category Options dialog or Series Options dialog to define the sort order of the category or series values and specify the number of the category or series values that will be displayed in the chart.

Adds a combo chart to the Primary Axis or Secondary Axis.

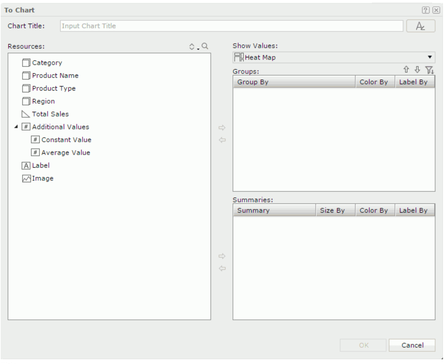
Moves the selected view element one level up.

Moves the selected view element one level down.

Adds a new pair of Y Axis and Radius for the bubble chart.

Removes the selected view element.

For heat map



Title

Specifies the title of the chart. The title is a special label bound with the chart. Though it can be positioned freely in a report, once you remove the chart from the report, the title will be removed too.

Specifies the font properties of the chart title.

Resources

Displays the resources that can be added to the chart.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

Launches the quick search toolbar to search for view elements.

Adds the selected field into the Groups or Summaries box.

Removes the selected field from the Groups or Summaries box.

Show Values

Displays Heat Map as the selected chart type.

Groups

Lists the fields used to group the data. There should be at least one group.

- **Color by**
  Specifies whether to color by a group. 0-n groups can be used as the color-by fields.
- **Label by**
  Specifies whether to show a group field in the innermost rectangle. 0-n groups can be used as the label-by fields.
-
- Moves the selected group field one level up.
-
- Moves the selected group field one level down.
-
- Opens the Group Options dialog to define the sort order of the group values and specify the number of the group values that will be displayed in the chart.
-
- Removes the selected group field.

Summaries

Lists the summaries used as size-by/color-by or displayed in the innermost rectangle.

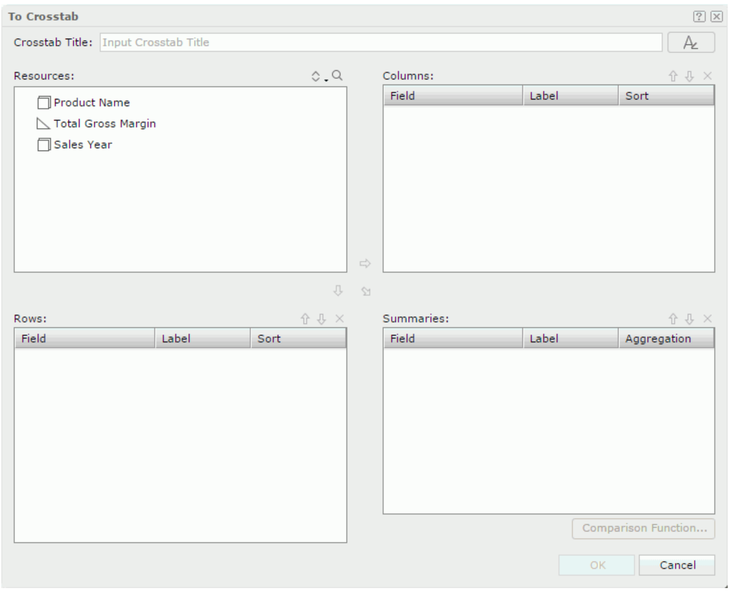
The summaries should match the groups. For example, if the groups level is A > B > C, the static summaries grouped by C can be inserted into the Summaries box, but the static summaries grouped by A, B or other fields cannot.

If no group is specified in the Groups box, you can insert any static summary. And its group-by field will be inserted into the Groups box automatically.

- **Size by**
  Specifies to size by one summary or none.
- **Color by**
  Specifies to color by one summary or none.
- **Label by**
  Specifies whether to show a summary in the innermost rectangle. 0-n summaries can be used as the label-by fields.

To Crosstab dialog

This dialog appears when you right-click a chart and then select To Crosstab on the shortcut menu or click Menu > Edit > To Crosstab. It allows you to specify settings for converting a chart into a crosstab.



Title

Specifies a title for the crosstab.

Specifies the font properties of the chart title.

- **Font**
  Lists all the available font faces that can be selected to apply to the title.
- **Font Style**
  Specifies the font style of the title. It can be one of the following: plain, bold, italic, and bold italic.
- **Size**
  Specifies the font size of the title.
- **Align**
  Specifies the position of the title to be left, right, center or justify.
- **Font Color**
  Specifies the font color of the title.
- **Background Color**
  Specifies the background color of the title.

Resources

Displays all the view elements used in the chart.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.
-
  Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
-
  Highlights the previous matched text.
-
  Highlights the next matched text.

Adds the selected group object to be displayed in the columns of the crosstab.

Adds the selected group object to be displayed in the rows of the crosstab.

Adds the selected aggregation object to be the summary field of the crosstab.

Columns/Rows

Lists the group objects that will be displayed on the columns/rows of the crosstab.

- **Field**
  Lists the group objects that you selected to display in the crosstab.
- **Label**
  Specifies the display names of the added group objects. By default these are blank and no names will be created for the group objects to label the columns/rows. You can click in the cells to edit them if required.
- **Sort**
  Specifies how the selected group objects will be sorted.

Summaries

Lists the aggregation/detail objects that will be the summary fields of the crosstab.

- **Field**
  Lists the aggregation/detail objects that you selected to display in the crosstab.
- **Label**
  Specifies the display names of the summary fields. By default these are blank and no names will be created for the objects to label the summaries. You can click in the cells to edit them if required.
- **Aggregation**
  Specifies the functions used to summarize data of the detail objects.
- **Comparison Function**
  Opens the Comparison Function dialog to add a comparison function as an aggregate for the crosstab.

Moves the selected view element one level up.

Moves the selected view element one level down.

Removes the selected view element.

OK

Applies the settings and closes the dialog.

Cancel

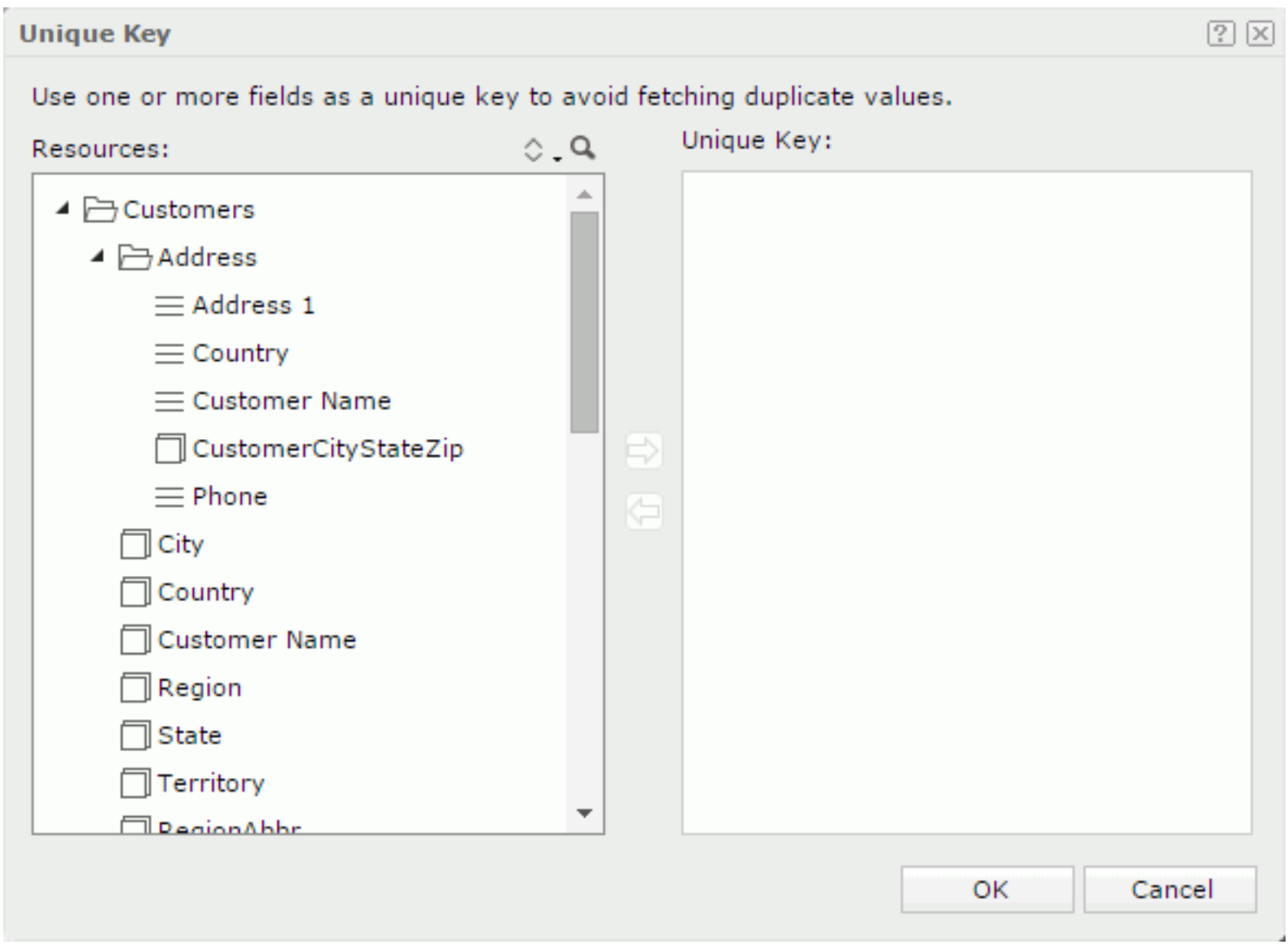
Cancels the settings and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Unique Key dialog

The dialog appears when you click the Incremental Fetch button in the Bind Data screen of chart in the Web Report Wizard, or click the button in the Insert Chart dialog or Chart Wizard. It helps you to specify a unique key for the real time chart.



Resources

Lists all the available data resources in the dataset used by the real time chart.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.
-
  Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
-
  Highlights the previous matched text.
-
  Highlights the next matched text.

Adds the selected fields to the Unique Key box.

Removes the selected from the Unique Key box if they are not required.

Unique Key

Lists the fields you have added as the unique key, which is used to guarantee there are no two records with the same unique key value in the real time chart.

When a real time chart is specified with a unique key, each time when the chart automatically updates itself, the data that has the same unique key value as the previous loaded data records will be filtered and only data with different unique key value will be added to the chart. For instance, if you add the fields Country and Product ID as the unique key of a real time chart, when a record with the product ID 1 in USA has already been loaded into the chart, no more records of this product ID in USA will be added to the real time chart because they have the same unique key value.

OK

Applies the settings and closes this dialog.

Cancel

Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Web Report Wizard

This wizard appears when you click New > Web Report on the JReport Console > Resources page, or click Menu > File > New Report or the New Report button on the Standard toolbar in Web Report Studio. It guides you through the process of creating a web report, and contains the following screens:

- Page screen
- Layout screen
- Bind Data screen
- Style screen

Displays the help document about this feature.

Back

Goes back to the previous screen.

Next

Goes to the next screen.

Save

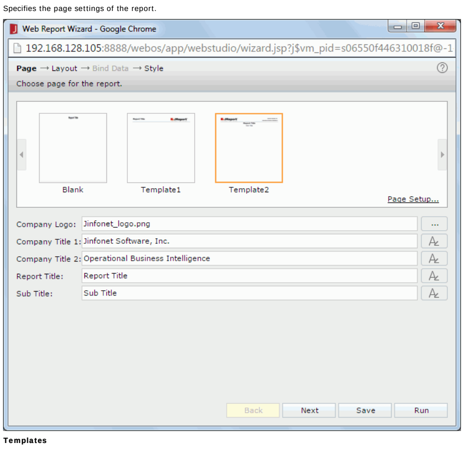
Saves the report to the server resource tree.

Run

Opens the report in Web Report Studio.

Page screen

Specifies the page settings of the report.



Templates

Specifies the template to be applied to the report.

- **Blank**
  Specifies to use the blank template.
- **Template1**
  Specifies to use Template1, in which you can specify the report title and company logo.
- **Template2**
  Specifies to use Template2, in which you can specify the company logo, company title, report title and sub title.

Page Setup

Opens the Page Setup dialog to specify the page properties.

Report Title

Specifies the display name of the report title.

Company Logo

Specifies the company logo image file.

Company Title 1

Specifies the display name of the company title 1.

Company Title 2

Specifies the display name of the company title 2.

Sub Title

Specifies the display name of the sub title.

Specifies the font properties for report title, sub title or company title.

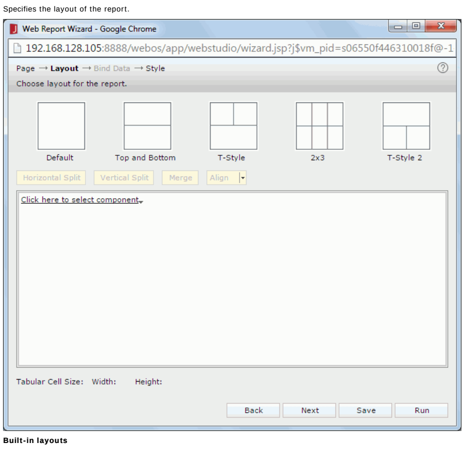
- **Font**
  Lists all the available font faces that can be selected to apply to the title.
- **Font Style**
  Specifies the font style of the title. It can be one of the following: plain, bold, italic, and bold italic.
- **Size**
  Specifies the font size of the title.
- **Align**
  Specifies the position of the title to be left, right, center or justify.
- **Font Color**
  Specifies the font color of the title.
- **Background Color**
  Specifies the background color of the title.

...

Opens the Insert Image dialog to customize the image for company logo.

Layout screen

Specifies the layout of the report.



Built-in layouts

Lists the built-in layouts.

- **Default**
  There is only a cell in the tabular.
- **Top and Bottom**
  There are two cells in the tabular. One is at the top, and the other is at the bottom. Each cell uses half the height of the tabular.
- **T-Style**
  There are two rows in the tabular. Each row uses half the height of the tabular. The top row has two cells with each using half the width of the row, and the bottom row has only a cell.
- **2×3**
  There are six cells in the tabular. Each cell uses one third the height and one third the width of the tabular.
- **T-Style 2**
  There are two rows in the tabular. Each row uses half the height of the tabular. The top row has only a cell, and the bottom row has two cells with each using half the width of the row.

Toolbar

The toolbar is enabled when a tabular cell in the below edit layout box is selected.

- **Horizontal Split**
  Splits the selected tabular cell into two cells horizontally.
- **Vertical Split**
  Splits the selected tabular cell into two cells vertically.
- **Merge**
  Merges the selected adjacent tabular cells that form a rectangular into one cell.
- **Align**
  Specifies how the component aligns in the tabular cell.
  - **Left**
    Aligns the component to the left of the tabular cell.
  - **Center**
    Aligns the component to the center of the tabular cell.
  - **Right**
    Aligns the component to the right of the tabular cell.

Edit layout box

Specifies the component to be inserted into the selected tabular cell.

- **Existing Components**
  Specifies a component from the ones existing in the open report to create the report.
- **New Components**
  Specifies the component you want to place in the tabular cell.
  - **Table**
    Creates a table in the tabular cell.
  - **Crosstab**
    Creates a crosstab in the tabular cell.
  - **Chart**
    Creates a chart in the tabular cell.
  - **Blank**
    Creates nothing in the tabular cell.

Tabular Cell Size

Displays the size of the selected tabular cell, in percent.

- **Width**
  Shows the width of the tabular cell.
- **Height**
  Shows the height of the tabular cell.

Bind Data screen

Specifies the data source and the fields to be displayed in each component selected in the Layout screen. This screen differs according to the following component types: table, crosstab, chart. When the component type is specified as Blank, the screen is disabled.

For table component

Specifies the data displayed in the table.

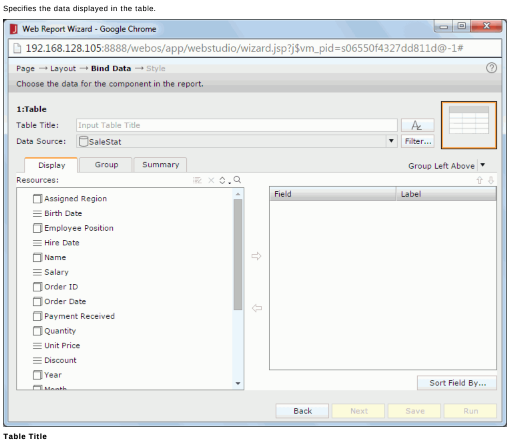


Table Title

Specifies the title of the table. The title is a special label bound with the table. Though it can be positioned freely in a report, once you remove the table from the report, the title will be removed too.

Specifies the font properties of the table title.

Data Source

Specifies the business view in the current catalog on which the table will be built.

Filter

Opens the Query Filter dialog to specify the filter you want to apply to the selected business view.

Table Type

Specifies the table type.

- **Group Above**
  Creates a table with group information above the detail panel.
- **Group Left**
  Creates a table with group information left to the detail panel.
- **Group Left Above**
  Creates a table with group information left above the detail panel.
- **Summary Table**
  Creates a table with only group and summary information.

Display tab

Specifies the fields to be displayed in the table.

- **Resources**
  Displays all the view elements in the selected business view. You can also create dynamic formulas to use in the table.
- Edits the selected dynamic resource.
- Removes the selected dynamic resource.
- Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.
  - **Predefined Order**
    Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
  - **Resource Types**
    Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
  - **Alphabetical Order**
    Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.
- Launches the quick search toolbar to search for view elements.
  - **Text box**
    The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
  - **X**
    Closes the quick search toolbar.
  - Lists more search options.
    - **Highlight All**
      Specifies whether to highlight all matched text.
    - **Match Case**
      Specifies whether to search for text that meets the case of the typed text.
    - **Match Whole Word**
      Specifies whether to search for text that looks the same as the typed text.
  - Highlights the previous matched text.
  - Highlights the next matched text.
- Adds the selected view element to be displayed in the table.
- Removes the selected view element.
- **Field**
  Lists the view elements that have been added to the table.
- **Label**
  Specifies the display names of the added view elements.
- Moves the selected view element one step up.
- Moves the selected view element one step down.
- **Sort Fields By**
  Opens the Custom Sort dialog to specify how to sort data in the table.

Group tab

Specifies the fields to group the data.

- **Resources**
  Displays all the available group objects you can use to group data in the table. You can also create dynamic formulas to use in the table.
- Edits the selected dynamic resource.
- Removes the selected dynamic resource.
- Sorts the group objects in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.
- Launches the quick search toolbar to search for group objects.
- Adds the selected group object as a group by field.
- Removes the selected group object.
- **Field**
  Lists all the group objects that have been added as the group by fields.
- **Sort**
  Specifies the sort order for groups at the specific group level.
  - **Ascend**
    Groups will be sorted in an ascending order (A, B, C).
  - **Descend**
    Groups will be sorted in a descending order (C, B, A).
  - **No Sort**
    Groups will be sorted in the original order in database.
  - **Custom Sort**
    Opens the Custom Sort dialog to set how groups will be sorted.
- Moves the selected group object one step up.
- Moves the selected group object one step down.

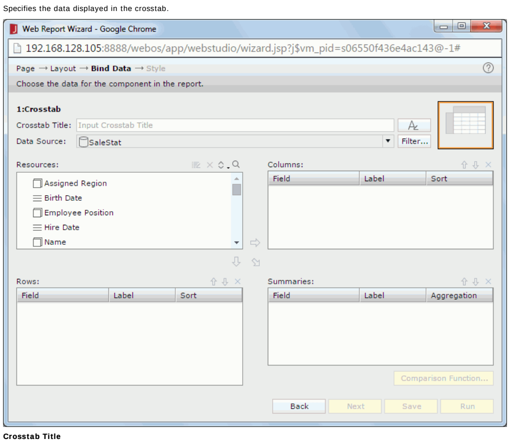
Summary tab

Specifies the fields on which to create summaries.

- **Resources**
  Displays all the available aggregation objects you can use to create summaries in the table. You can also create dynamic formulas and aggregations to use in the table.
- Edits the selected dynamic resource.
- Removes the selected dynamic resource.
- Sorts the aggregation objects in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.
- Launches the quick search toolbar to search for aggregation objects.
- Adds the selected aggregation object as the summary field.
- Removes the selected aggregation object.
- **Field**
  Lists the groups that have been added in the table and the aggregation objects added to summarize data in each group.
- **Row**
  Specifies to put the summary field in the header or footer row. If the summary is calculated on a group by field, it will be put in the group header or footer row of the corresponding group; if the summary is calculated on the whole dataset, it will be put in the table header or footer row. Available only when the table is Group Left type.
- **Column**
  Specifies to put the summary field in the specified detail column. Available only when the table is Group Left type.
- Moves the selected aggregation object one step up.
- Moves the selected aggregation object one step down.

For crosstab component

Specifies the data displayed in the crosstab.



Crosstab Title

Specifies the title of the crosstab. The title is a special label bound with the crosstab. Though it can be positioned freely in a report, once you remove the crosstab from the report, the title will be removed too.

Specifies the font properties of the crosstab title.

Data Source

Specifies the business view in the current catalog on which the crosstab will be built.

Filter

Opens the Query Filter dialog to specify the filter you want to apply to the selected business view.

Resources

Displays the elements in the selected business view. You can also create dynamic formulas and aggregations to use in the crosstab.

Edits the selected dynamic resource.

Removes the selected dynamic resource.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

Launches the quick search toolbar to search for view elements.

Adds the selected group object to be displayed on the columns of the crosstab.

Adds the selected group object to be displayed on the rows of the crosstab.

Adds the selected aggregation object and detail object to be the summary field of the crosstab.

Columns/Rows

- **Field**
  Lists the group objects that will be displayed on the columns/rows of the crosstab.
- **Label**
  Specifies the display names of the group objects. By default these are blank and no names will be created for the group objects to label the columns/rows. You can double-click the cells to edit them if required.
- **Sort**
  Specifies the sort order of the group objects.

Summaries

- **Field**
  Lists the aggregation/detail objects that you select to create summaries.
- **Label**
  Specifies the display names of the aggregation/detail objects. By default these are blank and no names will be created for the objects to label the summaries. You can click in the cells to edit them if required.
- **Aggregation**
  Specifies the functions used to summarize data of the detail objects.
- **Comparison Function**
  Opens the Comparison Function dialog to add a comparison function as an aggregate for the crosstab.

Moves the selected view element one level up.

Moves the selected view element one level down.

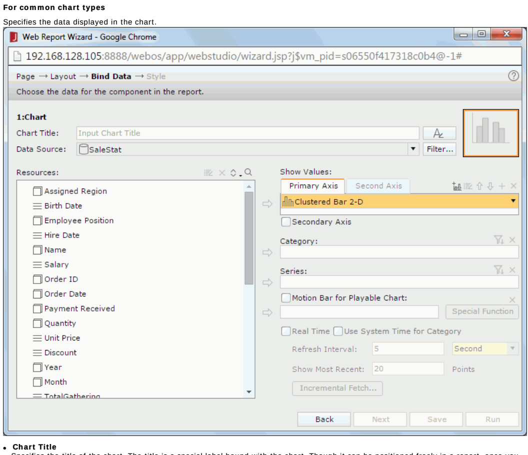
Removes the selected view element.

For chart component

The screen for the chart component varies with different chart types: common chart types, organization chart, heat map.

For common chart types

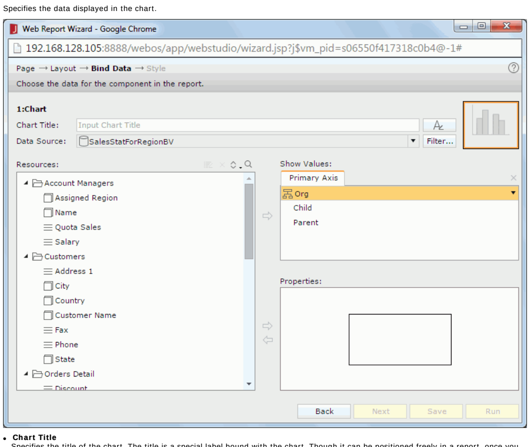
Specifies the data displayed in the chart.



- **Chart Title**
  Specifies the title of the chart. The title is a special label bound with the chart. Though it can be positioned freely in a report, once you remove the chart from the report, the title will be removed too.
- Specifies the font properties of the chart title.
- **Data Source**
  Specifies the business view in the current catalog on which the chart will be built.
- **Filter**
  Opens the Query Filter dialog to specify the filter you want to apply to the selected business view.
- **Resources**
  Displays the view elements in the selected business view. You can also create dynamic formulas and aggregations to use in the chart.
  - Edits the selected dynamic resource.
  - Removes the selected dynamic resource.
  - Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.
  - Launches the quick search toolbar to search for view elements.
- Adds the selected resource to be displayed in the chart.
- **Show Values**
  Lists the values you want to show in the chart. For a real time chart, the values you add must be of Numeric type and cannot be aggregation objects.
  - **Primary Axis**
    Adds a chart type to the primary axis.
  - **Secondary Axis**
    Adds a chart type to the secondary axis. Active only when the option Secondary Axis is checked.
  - **X Axis**
    Lists the value you want to show on the X axis of the bubble chart.
  - **Y Axis**
    Lists the value you want to show on the Y axis of the bubble chart.
  - **Radius**
    Lists the value you want to show as the bubble radius.
  - Adds a combo chart to the Primary Axis or Secondary Axis.
  - Opens the Edit Additional Value dialog to edit the selected additional value.
  - Moves the selected view element one level up.
  - Moves the selected view element one level down.
  - Adds a new pair of Y Axis and Radius for the bubble chart.
  - Removes the selected resource.
- **Secondary Axis**
  Specifies whether to show the secondary axis in the chart.
- **Category**
  Lists the group object that will be displayed on the category axis of the chart.
  For a real time chart, if no object is specified on the category axis, Use System Refresh Time will be automatically displayed in the Category text box, namely, the time at which the chart refreshes itself will be used as the category value.
- **Series**
  Lists the group object that will be displayed on the series axis of the chart. Not available to real time chart.
- Opens the Category Options dialog or Series Options dialog to define the sort order of the category or series values and specify the number of the category or series values that will be displayed in the chart.
- **Motion Bar for Playable Chart**
  Lists the group object you want to use as the motion field. A motion field can only be of Integer, Date or Time data type. Available to single bar, bench and bubble chart types only.
  - **Special Function**
    Available only when the motion field is of Date data type. Click it to define the special function.
    - **Field**
      Displays on which field the special function will be applied.
    - **Function**
      Specifies the special function to the field.
    - **OK**
      Accepts the special function settings and leaves the dialog.
    - **Cancel**
      Cancels the special function settings and leaves the dialog.
- **Real Time**
  Specifies to run the chart in real time mode, which means it will be updated automatically by using real time data. Available to single bar, bench, line, and area chart types only.
  - **Use System Time for Category**
    Specifies to use the time at which the chart refreshes itself as the category value.
  - **Refresh Interval**
    Specifies the time interval at which the chart will get data and refresh itself automatically.
  - **Show Most Recent N Points**
    Specifies the number of records that will be kept for the real time data on the chart.
  - **Incremental Fetch**
    Opens the Unique Key dialog to configure a unique key for the real time chart.

For organization chart

Specifies the data displayed in the chart.

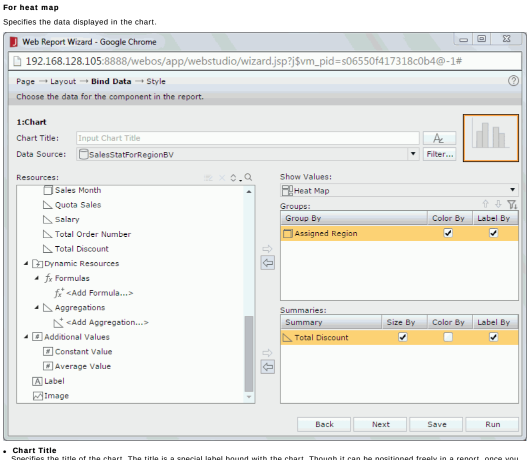


- **Chart Title**
  Specifies the title of the chart. The title is a special label bound with the chart. Though it can be positioned freely in a report, once you remove the chart from the report, the title will be removed too.
- Specifies the font properties of the chart title.
- **Data Source**
  Specifies the business view in the current catalog on which the chart will be built.
- **Filter**
  Opens the Query Filter dialog to specify the filter you want to apply to the selected business view.
- **Resources**
  Displays the resources that can be added to org chart. You can also create dynamic formulas and aggregations to use in the chart.
  - Edits the selected dynamic resource.
  - Removes the selected dynamic resource.
  - Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.
  - Launches the quick search toolbar to search for view elements.
- **Show Values**
  - **Primary Axis**
    Select Org from the chart type drop-down menu.
  - **Child**
    Adds a field from the Resources box which identifies the entity by selecting both the field and Child and then clicking .
  - **Parent**
    Adds a field from the Resources box which shows the "reporting to" relationship among the entity members, that is, which child field member reports to or belongs to which child field member, by selecting both the field and Parent and then clicking .
  - Removes the selected child or parent field.
- **Properties**
  The properties box presents a node model of the org chart. Data fields, labels and images can be inserted into the node as the information about the entity in the org chart, by using . By default all added objects are placed at the left top of the node, you need to adjust their positions and sizes in the node. You can also resize the node if required.

  To remove an object from the node, select it and then click .

For heat map

Specifies the data displayed in the chart.



- **Chart Title**
  Specifies the title of the chart. The title is a special label bound with the chart. Though it can be positioned freely in a report, once you remove the chart from the report, the title will be removed too.
- Specifies the font properties of the chart title.
- **Data Source**
  Specifies the business view in the current catalog on which the chart will be built.
- **Filter**
  Opens the Query Filter dialog to specify the filter you want to apply to the selected business view.
- **Resources**
  Displays the resources that can be added to the chart. You can also create dynamic formulas and aggregations to use in the chart.
  - Edits the selected dynamic resource.
  - Removes the selected dynamic resource.
  - Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.
  - Launches the quick search toolbar to search for view elements.
- Adds the selected field into the Groups or Summaries box.
- Removes the selected field from the Groups or Summaries box.
- **Show Values**
  Displays Heat Map as the selected chart type.
- **Groups**
  Lists the fields used to group the data. There should be at least one group.
  - **Color by**
    Specifies whether to color by a group. 0-n groups can be used as the color-by fields.
  - **Label by**
    Specifies whether to show a group field in the innermost rectangle. 0-n groups can be used as the label-by fields.
  - Moves the selected group field one level up.
  - Moves the selected group field one level down.
  - Opens the Group Options dialog to define the sort order of the group values and specify the number of the group values that will be displayed in the chart.
  - Removes the selected group field.
- **Summaries**
  Lists the summaries used as size-by/color-by or displayed in the innermost rectangle.

  The summaries should match the groups. For example, if the groups level is A > B > C, the static summaries grouped by C can be inserted into the Summaries box, but the static summaries grouped by A, B or other fields cannot.

  If no group is specified in the Groups box, you can insert any static summary. And its group-by field will be inserted into the Groups box automatically.
  - **Size by**
    Specifies to size by one summary or none.
  - **Color by**
    Specifies to color by one summary or none.
  - **Label by**
    Specifies whether to show a summary in the innermost rectangle. 0-n summaries can be used as the label-by fields.

Style screen

Specifies the style of the report.

Page Setup - Google Docs
File Edit View Insert Format Tools Table Help
Page Setup
Page Setup
Page Size
Margins
Header
Footer
Page Numbers
Columns
Section
Page Color
Watermark
Header and Footer
Header
Header
Header

Windows Media Properties dialog

The dialog appears when you right-click a Windows Media object and select Properties from the shortcut menu. It helps you to specify the properties of the Windows Media object and contains the following tabs:

- General tab
- Controls tab

OK

Applies the settings and closes this dialog.

Cancel

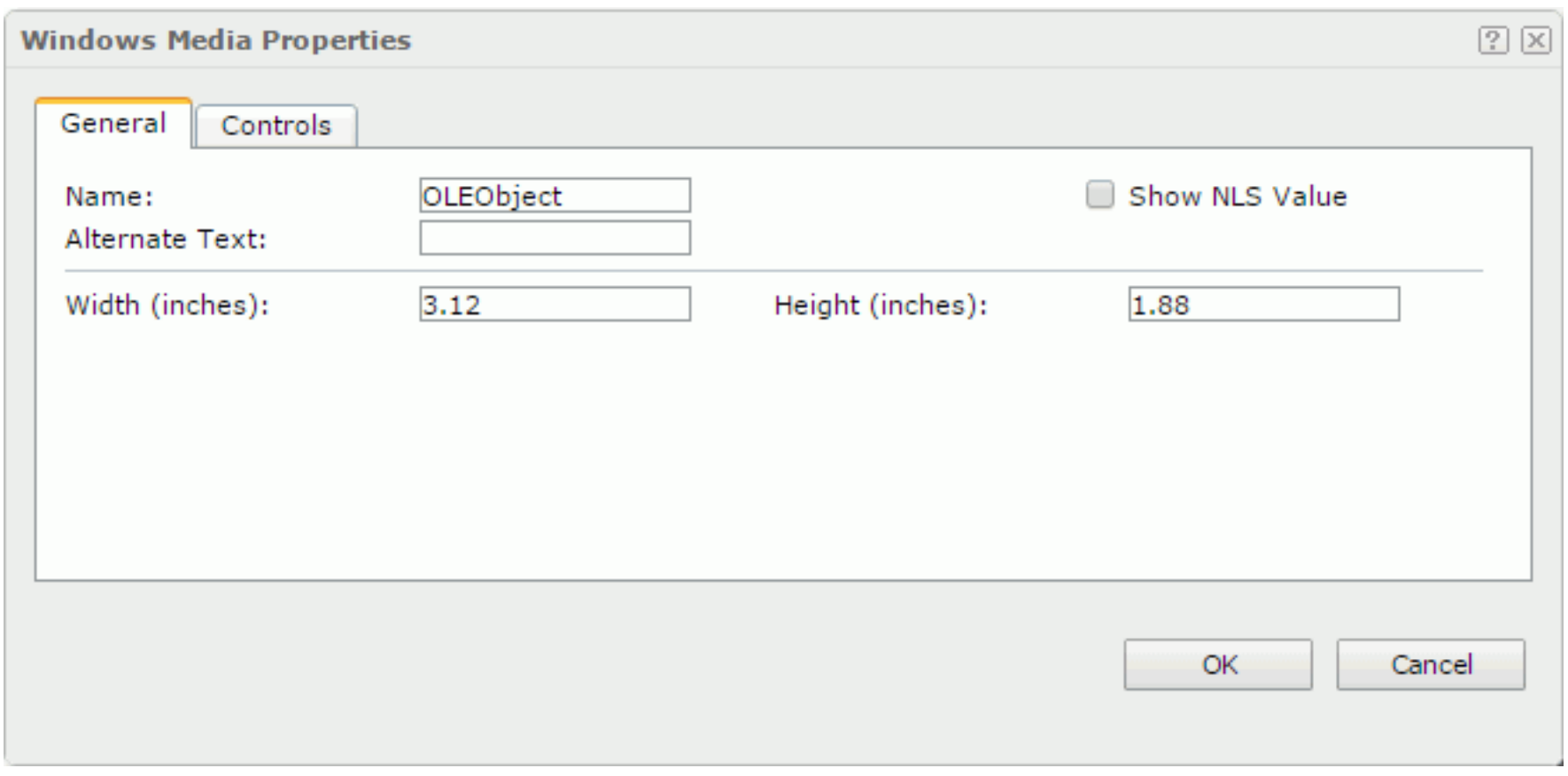
Cancels the settings and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

General tab

This tab shows some general information of the Windows Media.



Name

Specifies the display name of the Windows Media.

Show NLS Value

Specifies to show the translated name of the display name of the Windows Media in the Name text box if you have enabled the NLS feature and translated it.

If checked, this option takes effect only when the display name of the Windows Media is not modified.

Alternate Text

Specifies the alternate text which will be shown if the Windows Media cannot be displayed.

Width

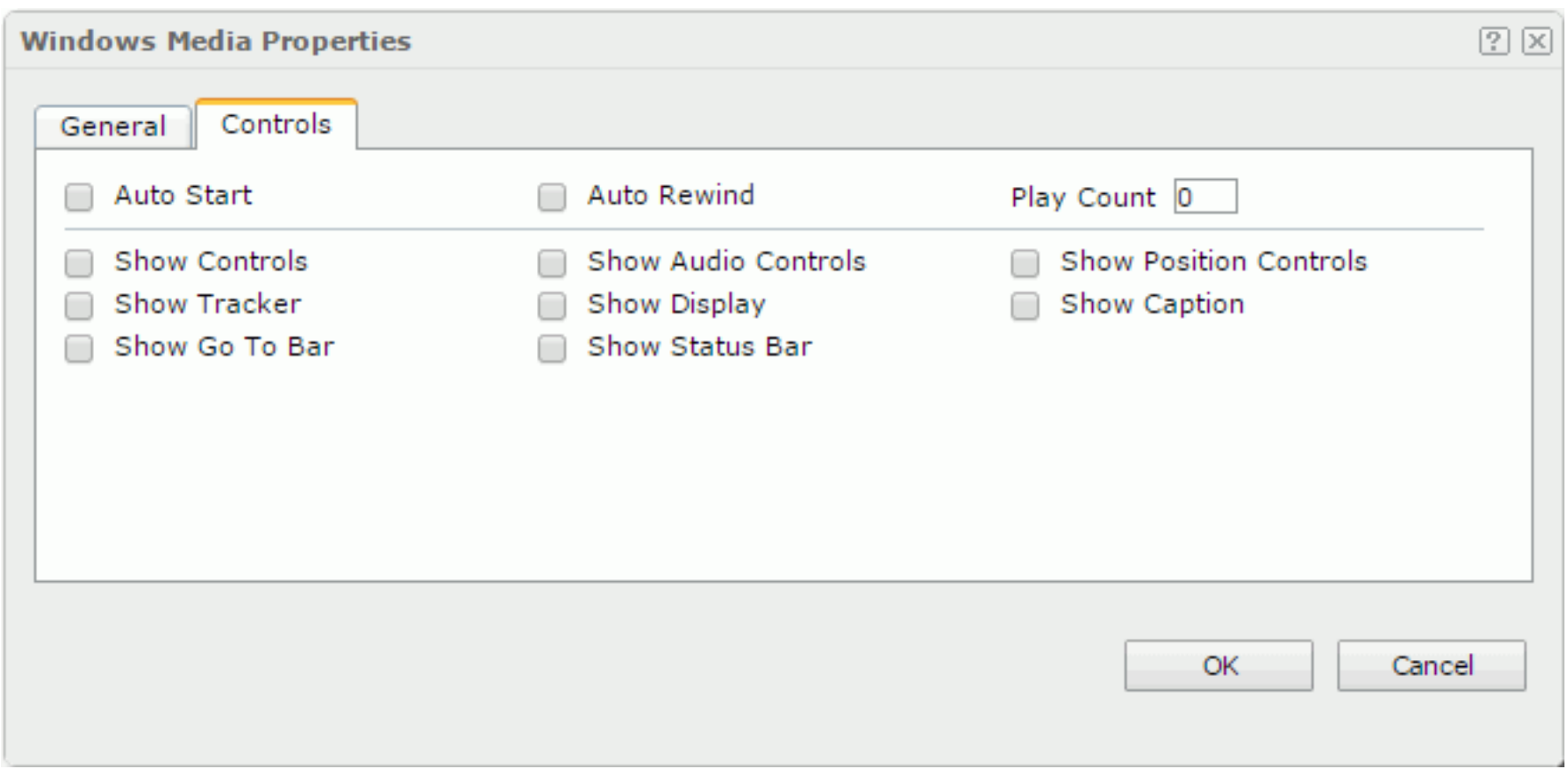
Specifies the width of the Windows Media.

Height

Specifies the height of the Windows Media.

Controls tab

This tab specifies the properties for the media file to control the options when playing the media file.



Auto Start

Specifies whether or not to start the media file automatically when the report is opened in Web Report Studio.

Auto Rewind

Specifies whether or not to rewind the media file automatically.

Play Count

Specifies the number of times you want to play this media file.

Show Controls

Specifies whether or not to show the Windows Media Player controls.

Show Audio Controls

Specifies whether or not to show the Mute Button and the Volume Slider on the Control Bar.

Show Position Controls

Specifies whether or not to show the Fast Forward, Fast Rewind, Skip Ahead, Skip Previous, and Preview buttons on the Control Bar.

Show Tracker

Specifies whether to show the Tracker Bar. It is only enabled when the Control Bar is visible.

Show Display

Specifies whether to show the Display Panel.

Show Caption

Specifies whether to show the Closed Captioning Displayed Panel.

Show Go To Bar

Specifies whether to show the Go To Bar.

Show Status Bar

Specifies whether to show the Status Bar.

JDashboard dialogs

While using JDashboard, you will go through the following dialogs. This section introduces each of these dialogs in alphabetical order. You can go into the links for details.

- Color Picker dialog
- Customize Component Title Bar dialog
- Customize Value dialog
- Edit Filter Control dialog
- Edit HTML dialog
- Edit Image dialog
- Edit Label dialog
- Edit Slider dialog
- Edit Special Field dialog
- Edit URL Frame dialog
- Enter Parameter Values dialog
- Enter Values dialog
- Export dialog
- Filter dialog
- Hyperlink dialog
- Insert Filter Control dialog
- Insert HTML dialog
- Insert Image dialog
- Insert Slider dialog
- Insert Special Field dialog
- Insert URL Frame dialog
- Open Dashboard dialog
- Open Document dialog
- Page Setup dialog
- Print dialog
- Printer Properties dialog
- Real-time Settings dialog
- Refresh Interval dialog
- Save As dialog
- Select Data Source dialog
- Select Parameter dialog
- Select Values dialog
- Share Parameters Setting dialog
- Themes dialog
- Unique Key dialog

Color Picker dialog

This dialog helps you to specify a color for an object. It contains the following tabs:

- Standard tab
- Custom tab

OK

Closes this dialog and returns the color value specified.

Cancel

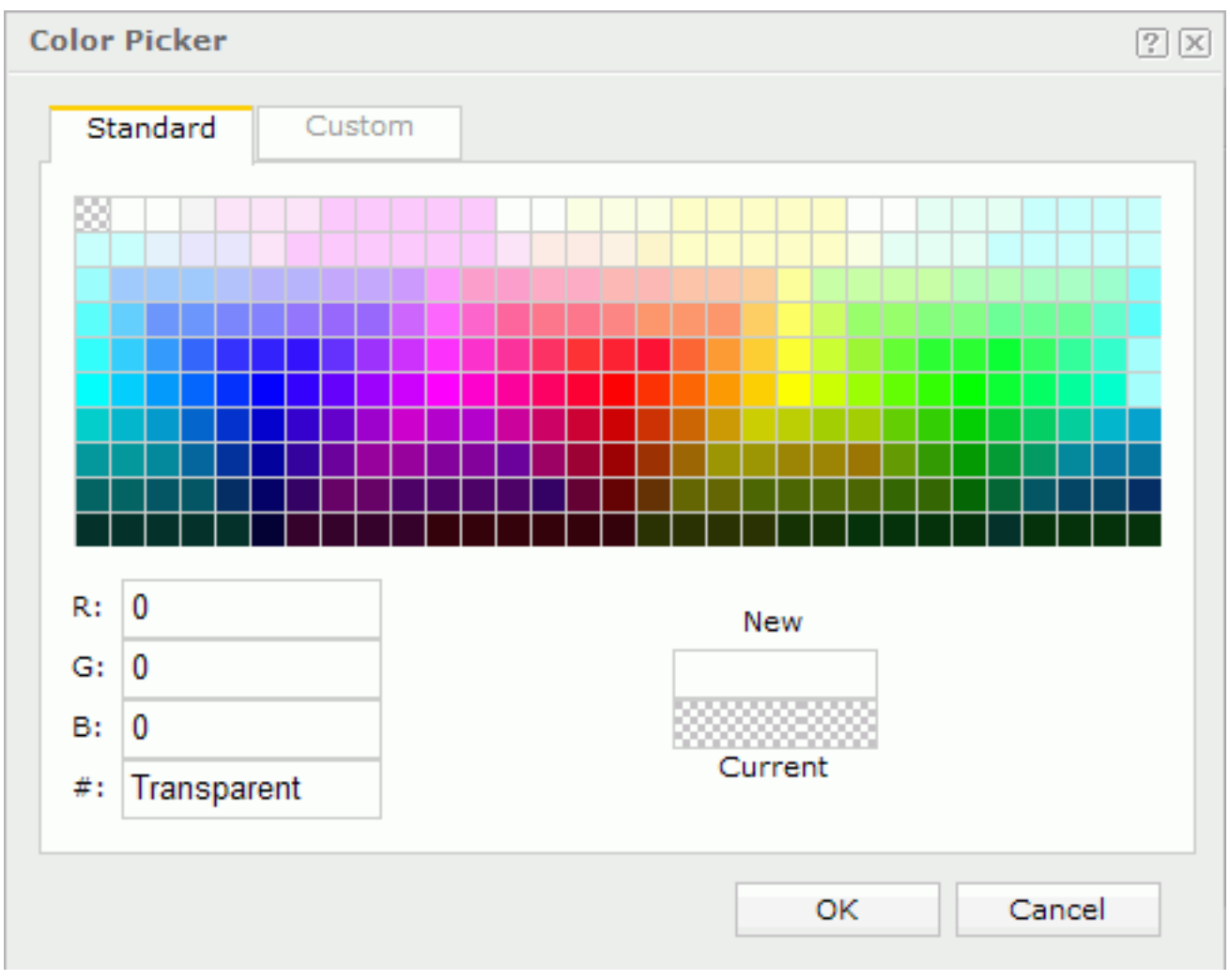
Closes this dialog without changing the color value.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Standard tab

This tab allows you to select one from the web safe colors.



Colors

Lists the colors for you to select one.

R

Specifies the amount of red in a color. The value is from 0 to 255.

G

Specifies the amount of green in a color. The value is from 0 to 255.

B

Specifies the amount of blue in a color. The value is from 0 to 255.

#

Specifies the hexadecimal value of the color.

New

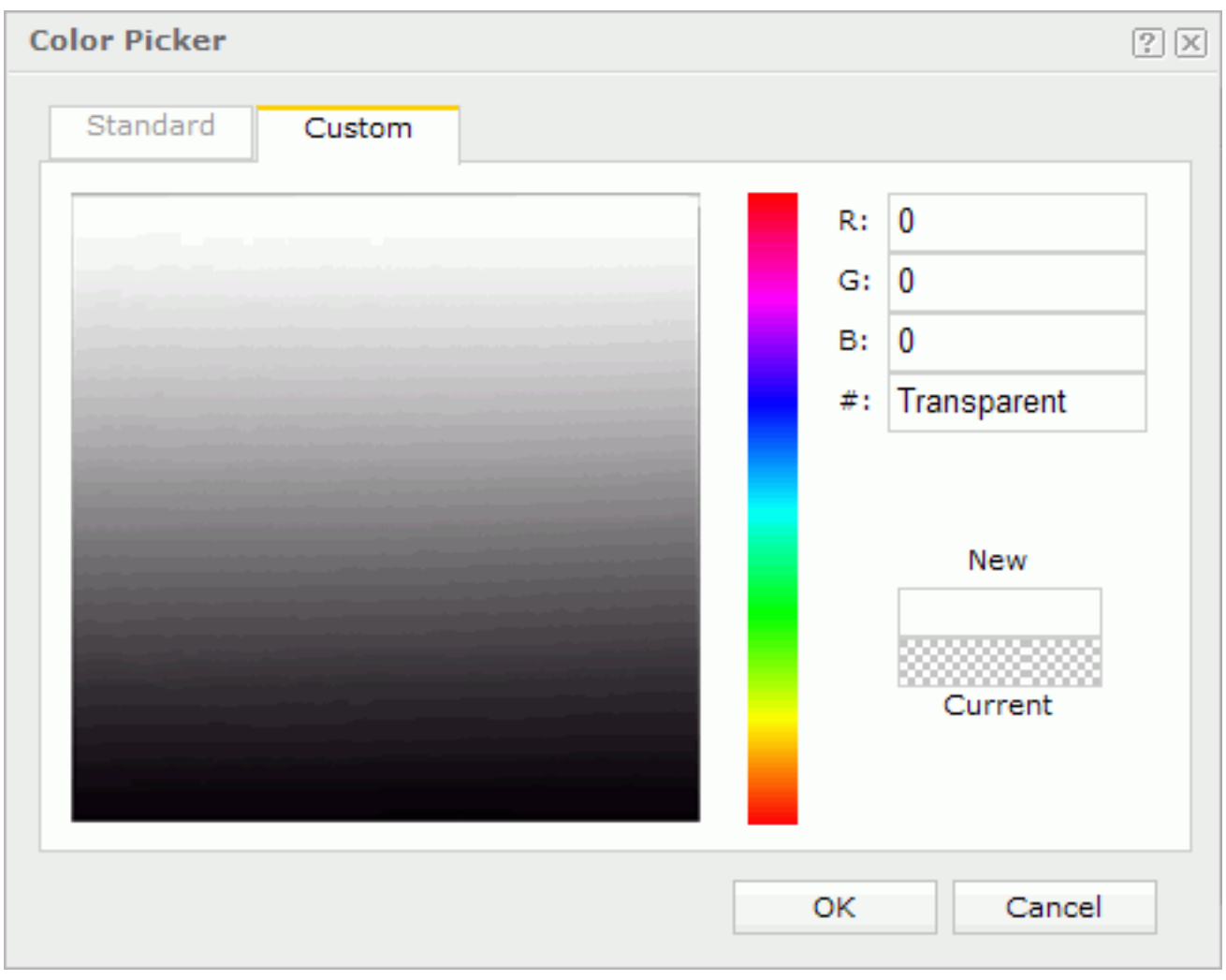
This part displays the color you select.

Current

This part always displays the original color. Clicking this part will restore the color to the original.

Custom tab

This tab allows you to customize the color within a wider range.



Color matrix

Select a color you want in the color matrix. The hue of the matrix is based on the color you define on the color bar.

Color bar

Specifies a color on the bar which will be the hue of the color matrix.

R

Specifies the amount of red in a color. The value is from 0 to 255.

G

Specifies the amount of green in a color. The value is from 0 to 255.

B

Specifies the amount of blue in a color. The value is from 0 to 255.

#

Specifies the hexadecimal value of the color.

New

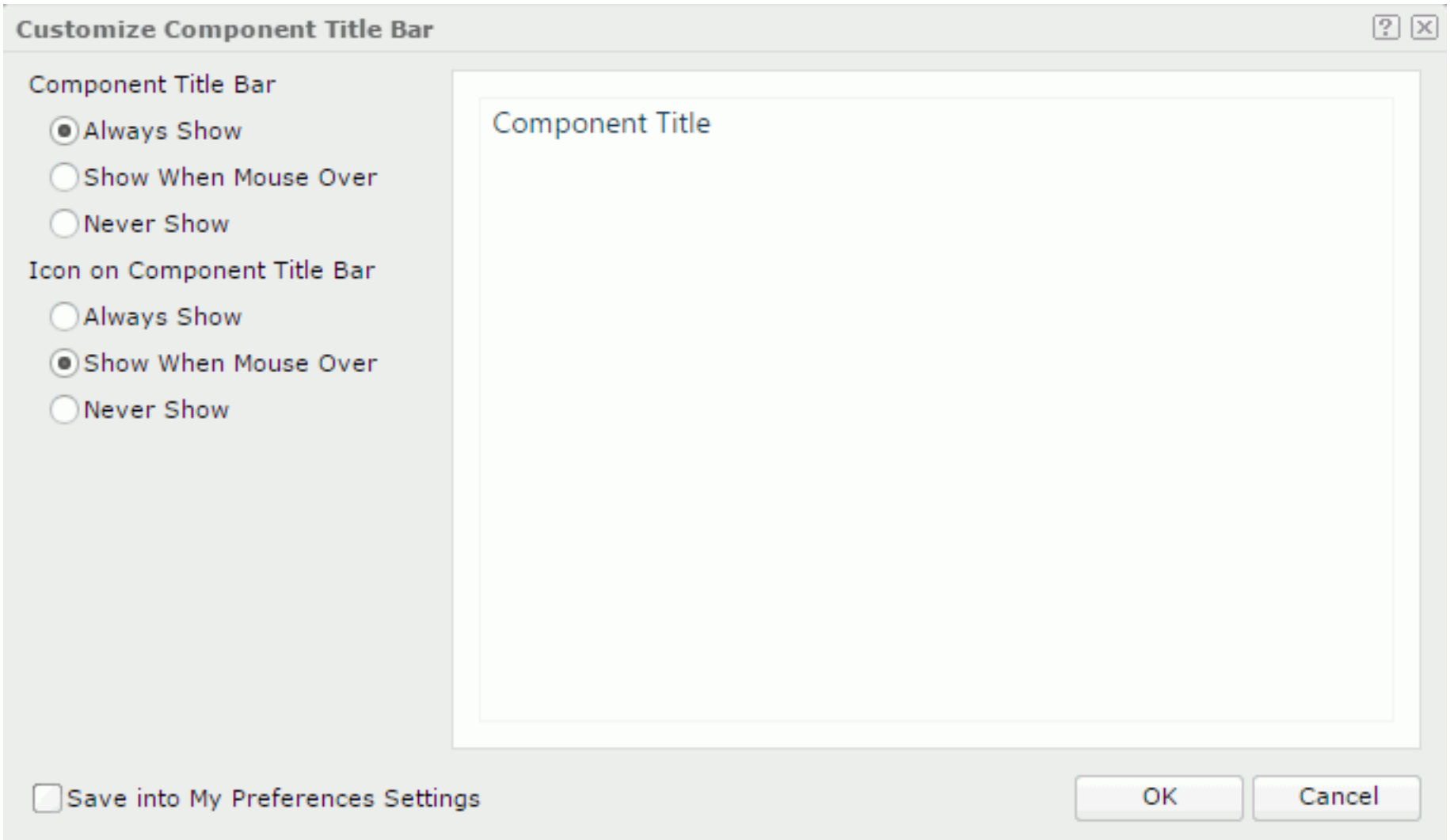
This part displays the color you select.

Current

This part always displays the original color. Clicking this part will restore the color to the original.

Customize Component Title Bar dialog

This dialog helps you to customize the way of showing component title bar and the icons on it.



Component Title Bar

Specifies the way of showing the component title bar.

Icon on Component Title Bar

Specifies the way of showing the icons on the component title bar.

Save into My Preferences Settings

Specifies whether to save the above settings into the current user's profile. If checked, after you click the **OK** button in the dialog, the above two settings will be applied to all the dashboards you open currently and later. If unchecked, after you click the **OK** button in the dialog, the above settings will be applied to the open dashboards during the current JDashboard session.

OK

Closes this dialog and applies the setting.

Cancel

Cancels the setting and closes this dialog.

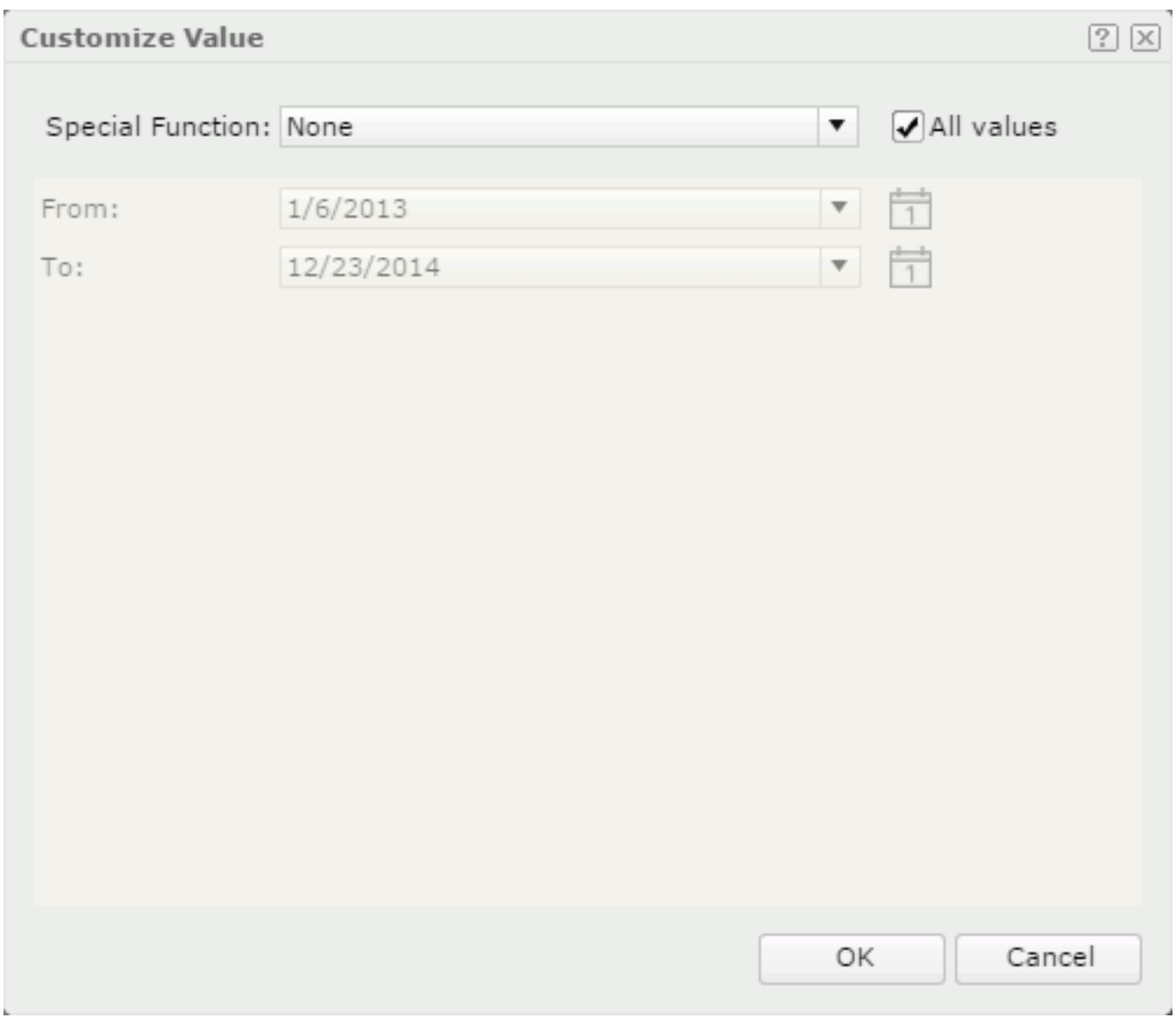
Displays the help document about this feature.

Ignores the setting and closes this dialog.

Customize Value dialog

The dialog appears when you specify the fields to bind to a slider and click the Customize button in the Insert Slider dialog or Edit Slider dialog. It helps you to customize the values of the selected fields available on the slider and varies with the slider type: Range of Values or Single Value.

If the slider type is Range of Values, options in the dialog are as follows.



Special Function

Specifies a special function from the drop-down list. Available to fields of Date/Time type.

All Values

This option is selected by default, which means that all the values of the selected field will be available on the slider. To customize the values, you need unselect this option.

From

Specifies the start value of the value range. You can select the value from the drop-down list or type it into the text box. For fields of Date/Time type, you can also use the calendar button to specify a value.

To

Specifies the end value of the value range. You can select the value from the drop-down list or type it into the text box. For fields of Date/Time type, you can also use the calendar button to specify a value.

OK

Closes this dialog and saves the customized values.

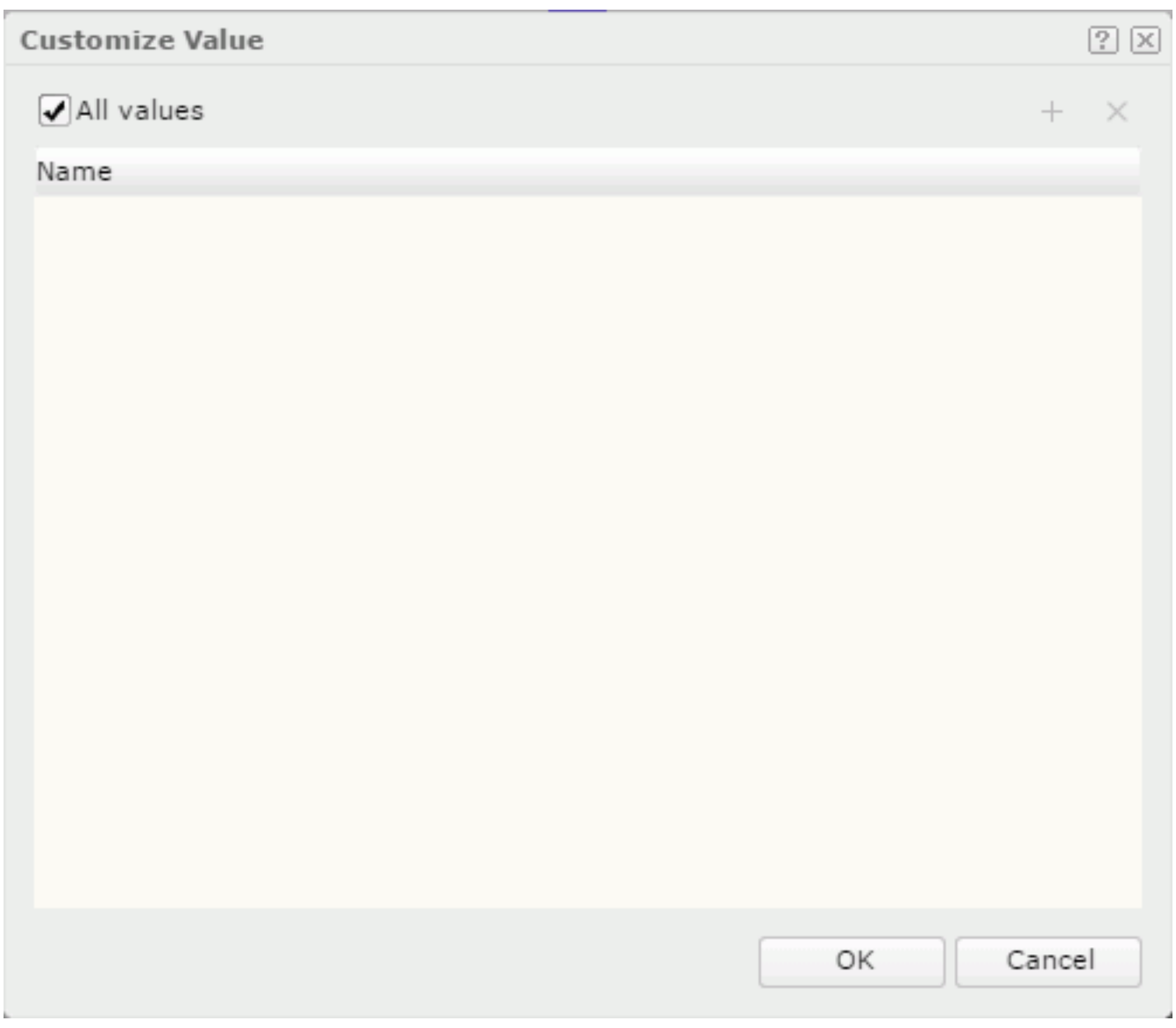
Cancel

Cancels customizing the values for the selected field.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

If the slider type is Single Value, options in the dialog are as follows.



Special Function

Specifies a special function from the drop-down list. Available to fields of Date/Time type.

All Values

This option is selected by default, which means that all the values of the selected field will be available on the slider. To customize the values, you need unselect this option.

Name

Specifies the values in the Name box.

Adds a value line to the Name box. You can select a value from the drop-down list or type a value into the text box.

Removes the selected value.

OK

Closes this dialog and saves the customized values.

Cancel

Cancels customizing the values for the selected field.

Displays the help document about this feature.

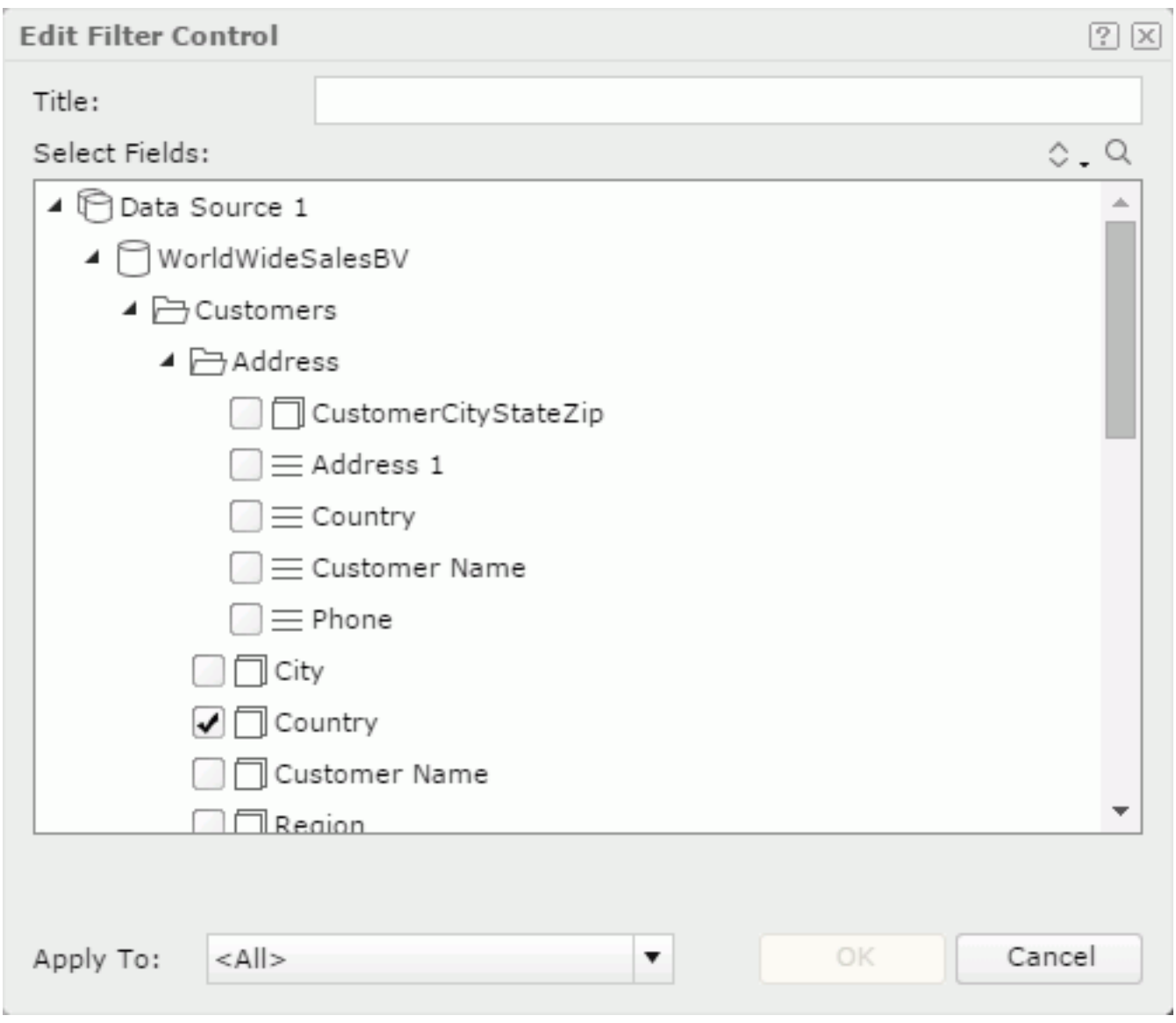
Ignores the setting and closes this dialog.

Related topics:

- Filtering component data

Edit Filter Control dialog

This dialog appears after you click on the title bar of a filter control and then select Edit Setting. It helps you to edit the filter control.



Title

Specifies a title for the filter control.

Select Fields

Specifies the fields to bind to the filter control. All the selected fields should be of the same data type. All uncomparable data type fields cannot be bound to filter controls, such as Binary, Blob, Clob, Longvarchar, Longvarbinary, and Varbinary.

The common usage is to select one field, and then based on the field to filter the data of the components created from the same data source as the field.

To filter components using different data sources, choose a common field all the data sources contain and select the field in all the data sources.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.
-
  Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
-
  Highlights the previous matched text.
-
  Highlights the next matched text.

Apply To

Specifies the data components from the drop-down list to apply the filter control to. <All> means all data components involving the selected fields in the dashboard.

OK

Closes this dialog and applies the changes to the filter control.

Cancel

Cancels the edition and closes this dialog.

Displays the help document about this feature.

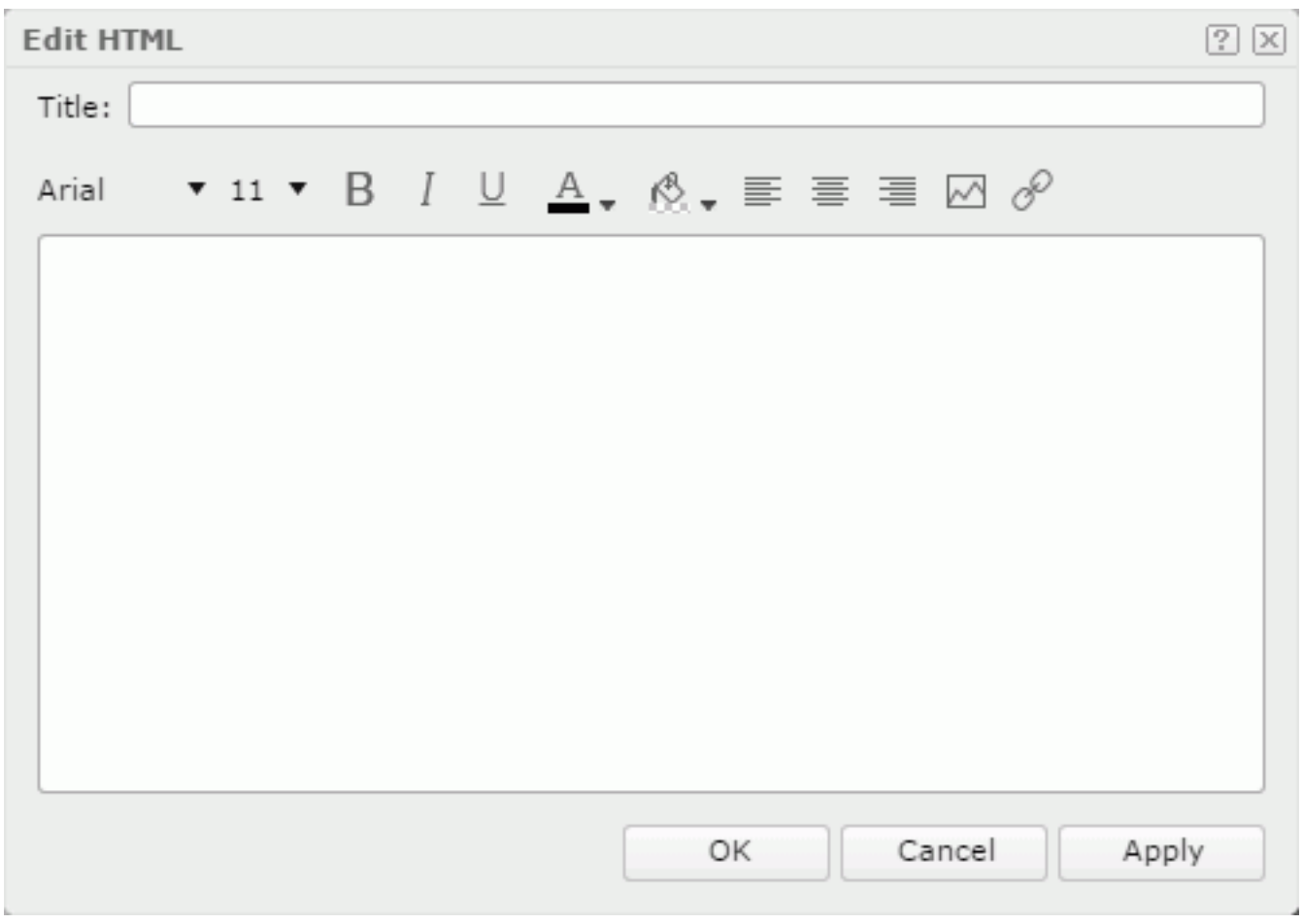
Ignores the setting and closes this dialog.

Related topics:

- Filtering component data

Edit HTML dialog

This dialog appears after you click on the title bar of an HTML component and then select Edit Setting. It helps you to edit the title and contents of the HTML component.



Title

Specifies a title for the HTML component.

HTML text box

Type text in the box directly. You can make use of the buttons above the text box to format the text, insert images, and create hyperlinks.

Here are the buttons:

- Arial **Font Face**
  Specifies the font of the text from the drop-down list.
- 11 **Font Size**
  Specifies the font size of the text from the drop-down list.
- B
  Specifies whether to make the text bold.
- I
  Specifies whether to make the text italic.
- U
  Specifies whether to underline the text.
- A **Font Color**
  Specifies the color of the text.

  To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range.
- **Background Color**
  Specifies the background color of the text.

  To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range.
- **Align Left**
  Specifies to align the text to the left.
- **Align Center**
  Specifies to align the text in the center.
- **Align Right**
  Specifies to align the text to the right.
- **Insert Image**
  Opens the Insert Image dialog for inserting an image into the HTML component.
- **Hyperlink**
  Opens the Hyperlink dialog to create a hyperlink on the selected object.

OK

Closes this dialog and applies the changes to the HTML component.

Cancel

Cancels the changes and closes this dialog.

Apply

Applies the changes to the HTML component without exiting the dialog.

?

Displays the help document about this feature.

x

Ignores the setting and closes this dialog.

Related topics:

- Inserting an HTML component

Edit Image dialog

This dialog appears when you hover the cursor on an image in the dashboard header and then click the button in the appearing dotted rectangle. It helps you to select another image to replace the current image.

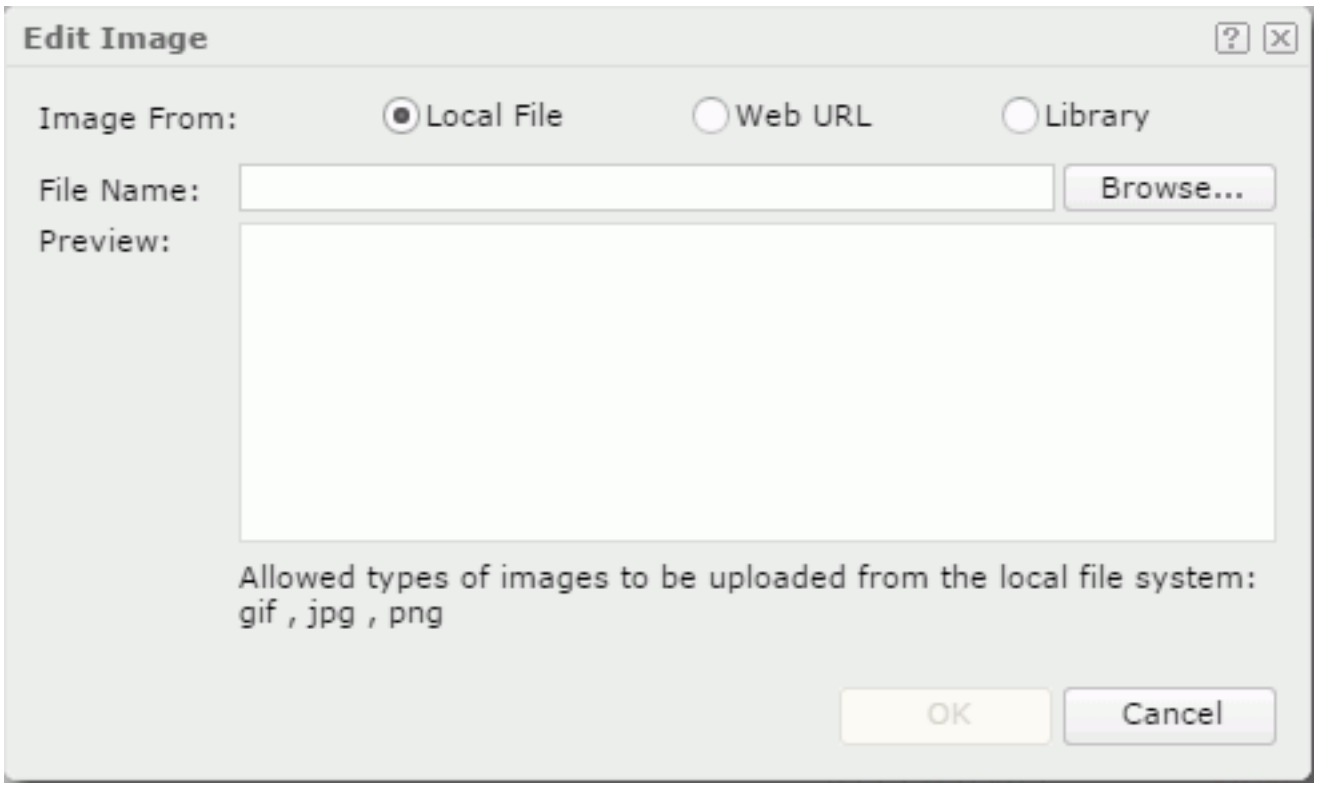


Image From

Specifies the source of the image file.

- **Local File**
  Specifies to use an image from the local file system.
  - **File Name**
    Specifies the path and name of the image file. You can click the Browse button to locate the image file.
- **Web URL**
  Specifies to use an image via URL.
  - **File URL**
    Specifies the URL of the image file.

    Note: If your JReport Server is in an intranet which requires a proxy, to successfully access the image via URL, you need to add the parameters `-Dhttp.proxyHost=XXX -Dhttp.proxyPort=XX` to the server's startup file JRServer.bat, which is located in `<install_root>\bin`.
- **Library**
  Specifies to use an existing image.
  - **My Pictures**
    The My Pictures folder is a virtual location where JReport Server stores the images that have once been inserted into JDashboard. Select the one you want to use.

Preview

Displays a preview of the selected image.

OK

Closes this dialog and use the selected image to replace the current image.

Cancel

Cancels the change of the image and closes this dialog.

Displays the help document about this feature.

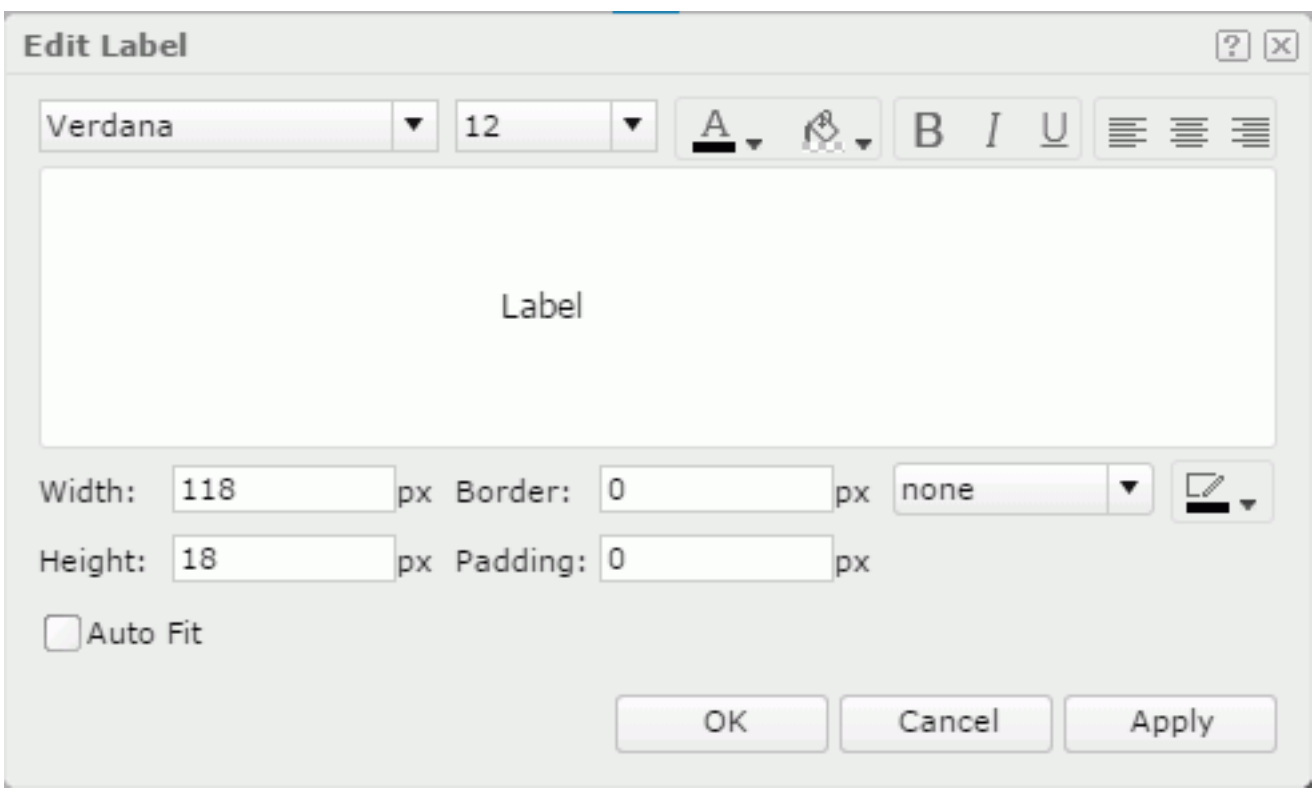
Ignores the setting and closes this dialog.

Related topics:

- Inserting an image

Edit Label dialog

This dialog appears when you hover the cursor on a label and then click the button in the appearing dotted rectangle. It helps you to edit the properties of the label such as font, size, color, border, the way of aligning text, and so on.



Font Face

Specifies the font of the text from the drop-down list.

Font Size

Specifies the font size of the text from the drop-down list.

Font Color

Specifies the color of the text.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range.

Background Color

Specifies the background color of the label.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range.

Specifies whether to make the text bold.

Specifies whether to make the text italic.

Specifies whether to underline the text.

Align Left

Specifies to align the text to the left.

Align Center

Specifies to align the text in the center.

Align Right

Specifies to align the text to the right.

Preview box

Displays a preview of the label with the specified properties.

Width

Specifies the width of the label.

Height

Specifies the height of the label.

Border

Specifies the border properties of the label.

- **Border Width**
  Specifies the width of the border.
- **Border Line**
  Specifies the line style of the border from the drop-down list.
- **Border Color**
  Specifies the color of the border.

  To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range.

Padding

Specifies the space between the text and the borders of the label.

Auto Fit

Specifies whether to adjust the width and height of the label according to the contents.

OK

Closes this dialog and applies the properties to the label.

Cancel

Cancels the edition and closes this dialog.

Apply

Applies the properties but does not close the dialog.

Displays the help document about this feature.

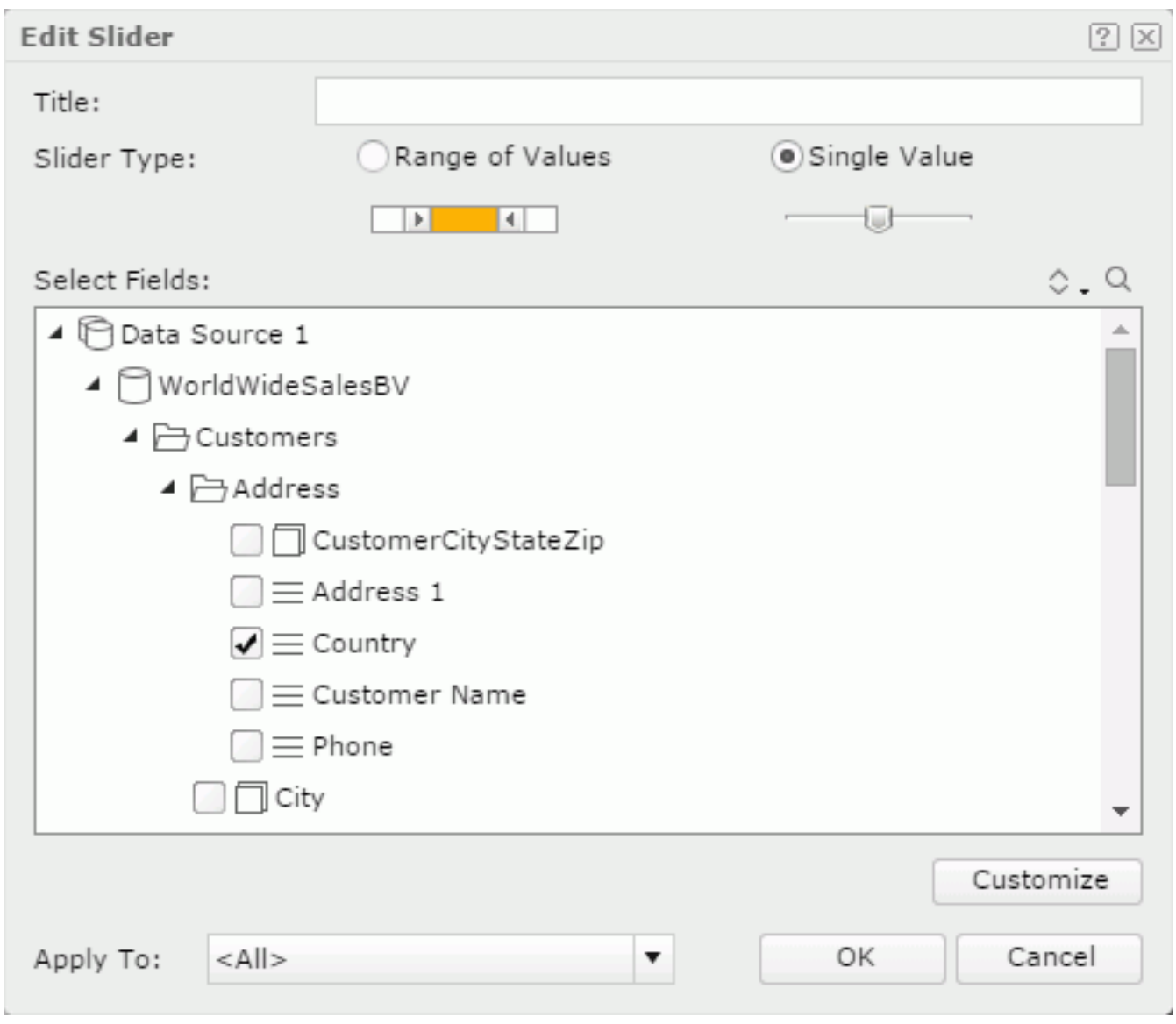
Ignores the setting and closes this dialog.

Related topics:

- Inserting a label

Edit Slider dialog

This dialog appears after you click on the title bar of a slider and then select Edit Setting. It helps you to edit the slider.



Title

Specifies a title for the slider.

Slider Type

Specifies whether the slider is used to specify a range of values or to specify a single value:

- **Range of Values**
  If selected, you can select a range of values at runtime.
- **Single Value**
  If selected, you can select one value at runtime. Not available for numeric data type fields.

Select Fields

Specifies the fields to bind to the slider. All the selected fields should be of the same data type.

The common usage is to select one field, and then based on the field to filter the data of the components created from the same data source as the field.

To filter components using different data sources, choose a common field all the data sources contain and select the field in all the data sources.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.
-
  Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
-
  Highlights the previous matched text.
-
  Highlights the next matched text.

Customize

Opens the Customize Value dialog for you to customize the values of the selected field.

Apply To

Specifies the data components from the drop-down list to apply the slider to. <All> means all data components involving the selected fields in the dashboard.

OK

Closes this dialog and applies the changes to the slider.

Cancel

Cancels the edition and closes this dialog.

Displays the help document about this feature.

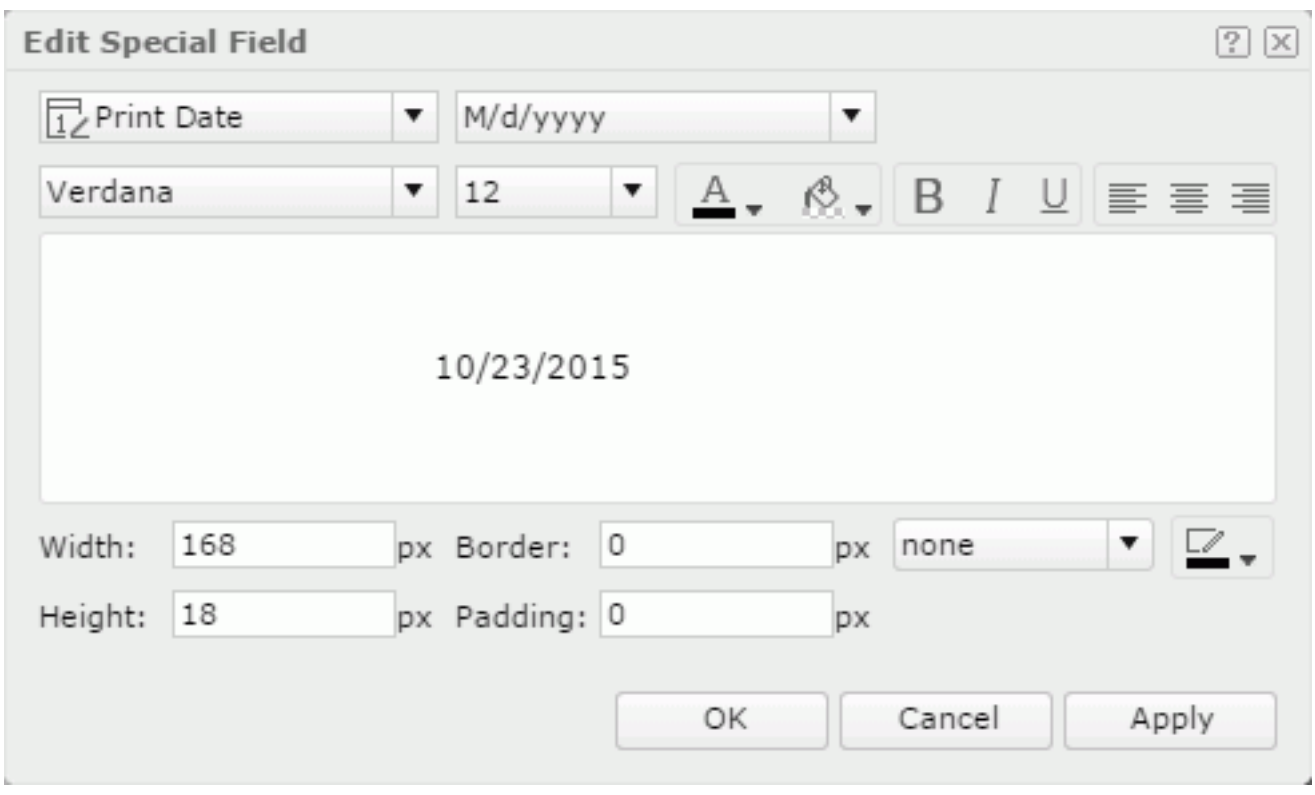
Ignores the setting and closes this dialog.

Related topics:

- Filtering component data

Edit Special Field dialog

This dialog appears when you hover the cursor on a special field and then click the button in the appearing dotted rectangle. It helps you to edit the properties of the special field such as font, size, color, border, the way of aligning text, and so on.



Field Type

The current special field is selected by default. If it is not User Name, a date or time format drop-down list will be displayed on the right for you to choose one.

Font Face

Specifies the font of the text from the drop-down list.

Font Size

Specifies the font size of the text from the drop-down list.

Font Color

Specifies the color of the text.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range.

Background Color

Specifies the background color of the special field.

To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range.

Specifies whether to make the text bold.

Specifies whether to make the text italic.

Specifies whether to underline the text.

Align Left

Specifies to align the text to the left.

Align Center

Specifies to align the text in the center.

Align Right

Specifies to align the text to the right.

Preview box

Displays a preview of the special field with the specified properties.

Width

Specifies the width of the special field.

Height

Specifies the height of the special field.

Border

Specifies the border properties of the special field.

-

Border Width
Specifies the width of the border.
-

Border Line
Specifies the line style of the border from the drop-down list.
-
Border Color
Specifies the color of the border.

  To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range.

Padding

Specifies the space between the text and the borders of the special field.

OK

Closes this dialog and applies the properties to the special field.

Cancel

Cancels the edition and closes this dialog.

Apply

Applies the properties but does not close the dialog.

Displays the help document about this feature.

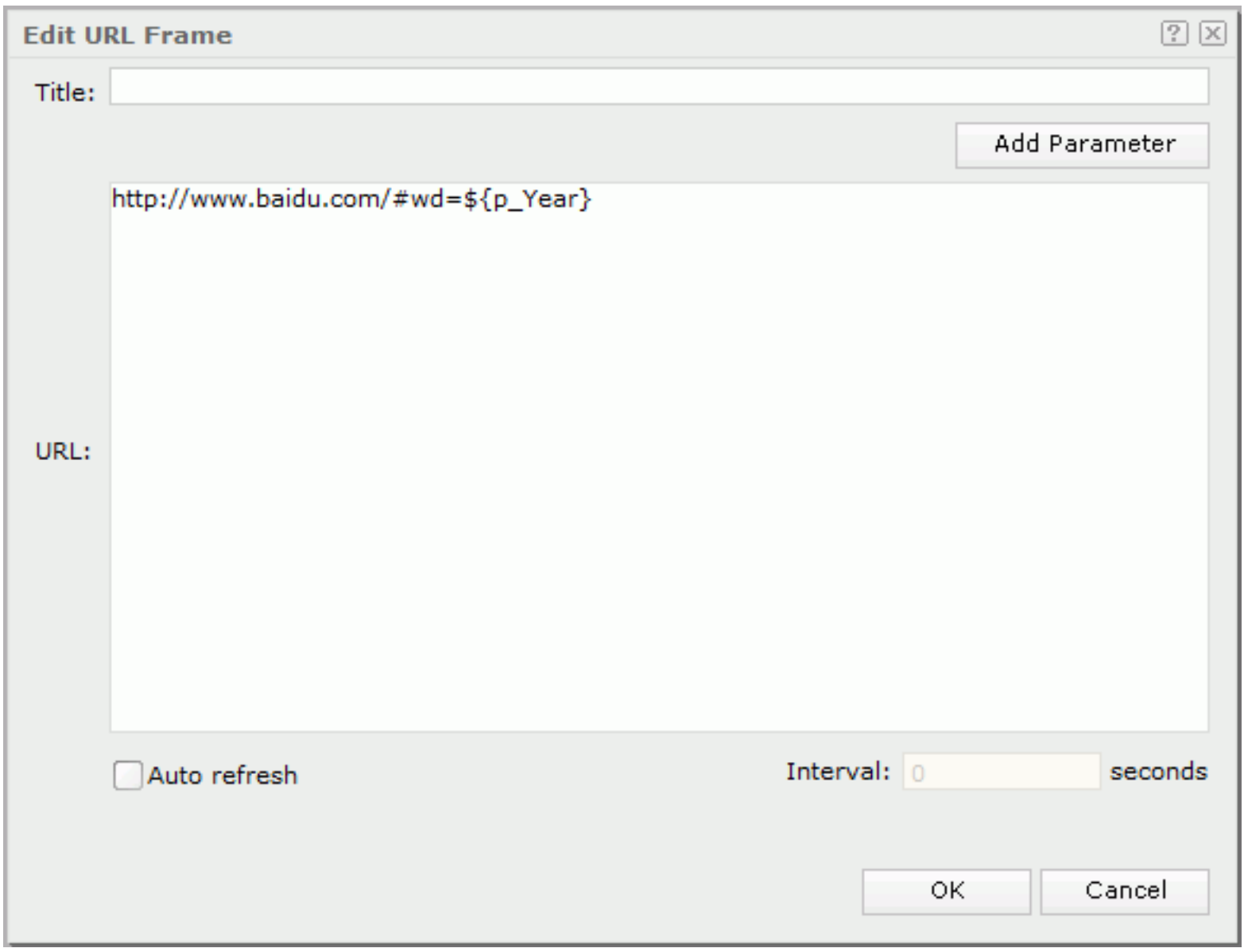
Ignores the setting and closes this dialog.

Related topics:

- Inserting a special field

Edit URL Frame dialog

This dialog appears when you click on the title bar of a web page window and then select Edit Setting. It helps you to edit the URL.



Title

Specifies the title of the frame that holds the web page.

URL

Specifies the URL of the web page you would like to show in the dashboard.

- **Add Parameter**
  Opens the Select Parameter dialog to insert a parameter into the URL to compose a dynamic URL if a data source has already been specified, or the Select Data Source dialog to select the data source that contains the required parameter if no data source has been selected.

Auto refresh

If checked, the specified web page will be refreshed periodically.

- **Interval**
  Specifies the time interval between two refreshes.

OK

Closes this dialog and applies the changes.

Cancel

Cancels the edition and closes this dialog.

Displays the help document about this feature.

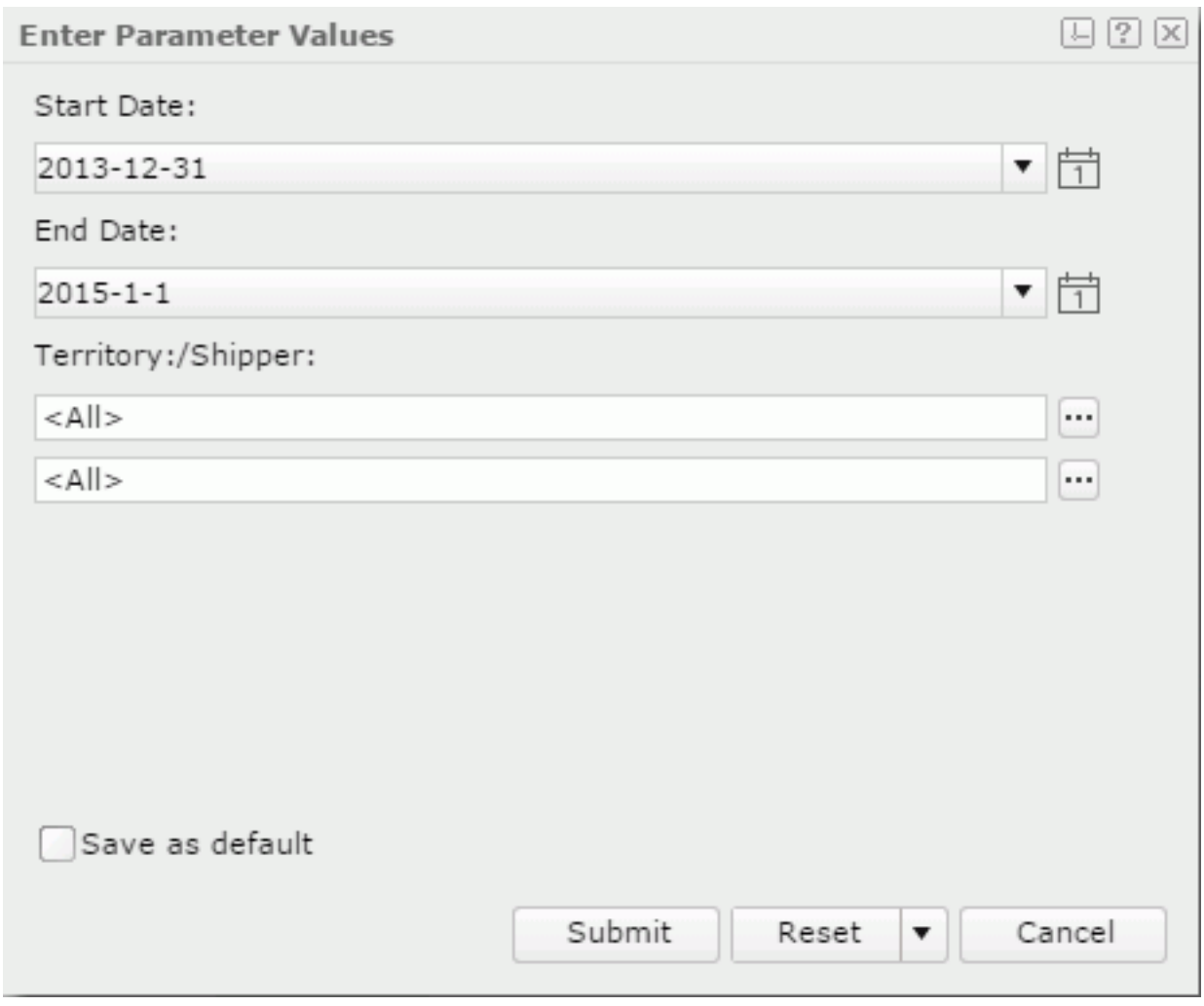
Ignores the setting and closes this dialog.

Related topics:

- Inserting a third party gadget via URL

Enter Parameter Values dialog

This dialog appears after you click on the side bar. In this dialog, all the parameters used in the current dashboard are listed. Same-name parameters are allowed when they are from different library components. You can specify the values to rerun the dashboard.



Parameters

Displays parameters of all the library components in the dashboard with their default values.

If you have not yet set the default values for the parameters in JDashboard using the save as default action, or if any of the parameters cannot find a matched validated value from the saved default values, all the parameters will use their default values specified in the parameters' definition as the initial values.

Save as default

Saves current parameter values as the default values for all the library components in the dashboard. It is a user-dashboard level setting.

If later you change the parameter sharing setting in the dashboard, the saved default values will be cancelled and the default values specified in the parameters' definition will be used.

JDashboard remembers the last-time saved result of a dashboard and displays the exact result the next time the dashboard is opened. In the case a dashboard user sets USA as the default parameter value and then filters the result with Canada and saves the dashboard, the next time the dashboard will run with Canada, though the saved default parameter value is still kept as USA.

This option is an action and it takes effect after you click Submit in the dialog. Its initial status is always unselected.

This option is available when Enable Setting Default Parameter Values For Dashboard is checked in the Profile > Customize Server Preferences > Advanced tab.

Submit

Closes this dialog and applies the specified values to run the dashboard.

Reset

When Save as default is available in the dialog, the Reset button contains a text part and a triangle icon. You can choose to reset the values to either of the following by clicking the triangle. If you click the text part of the button directly, the values will be reset to the original default values.

- **Original Default Values**
  The default values defined in the parameters' definition.
- **User Defined Default Values**
  The default values specified using the save as default action.

When Save as default is unavailable in the dialog, the Reset button contains only the text part. Clicking it will reset the values to the original default values.

Cancel

Cancels changing the parameter values and closes this dialog.

Use Saved Values

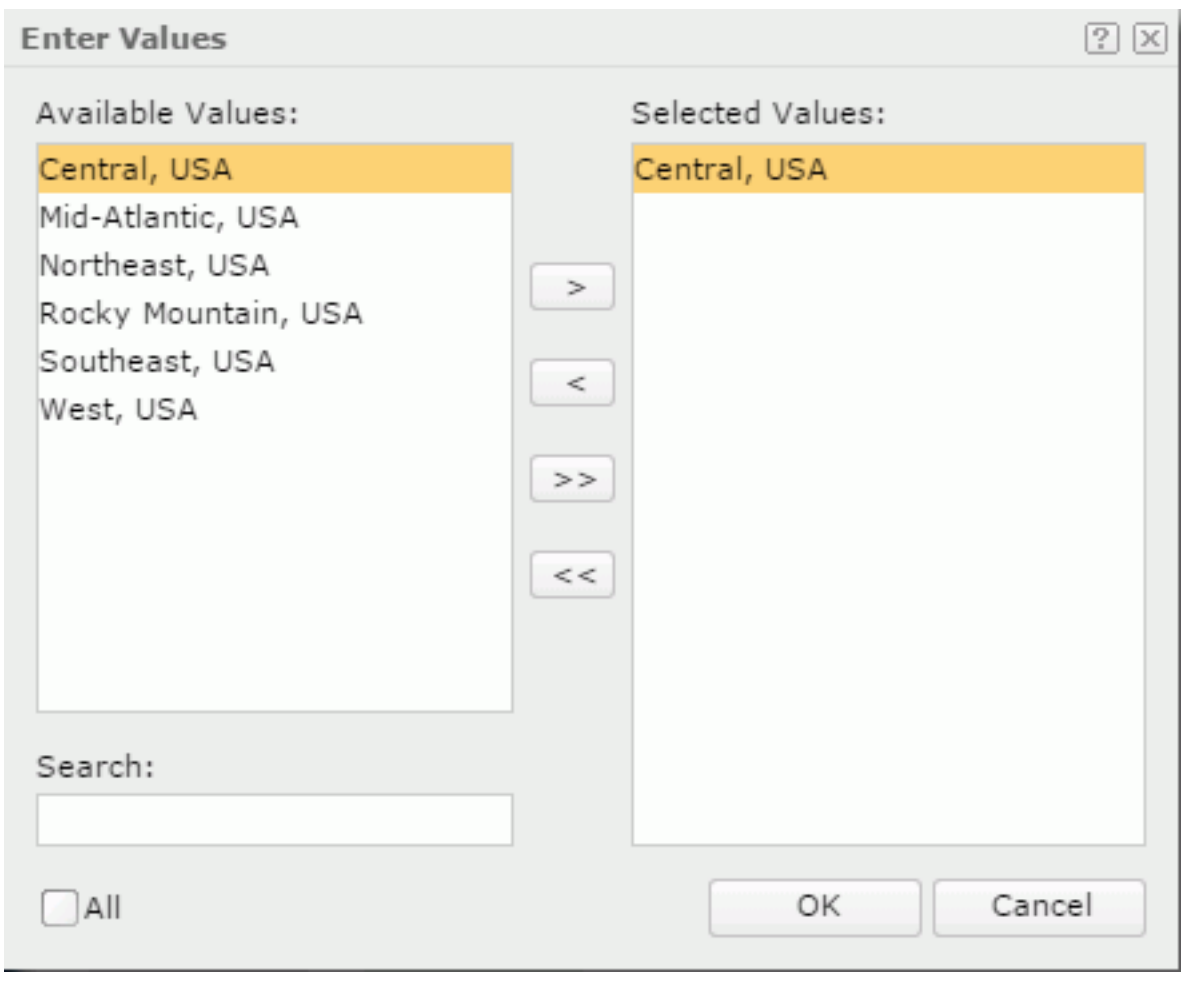
If it is available, you can select the previously saved parameter values to apply to the dashboard and save parameter values for reuse later. For details, see Saving parameter values for reuse.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Enter Values dialog

The dialog appears when you click the button ... while specifying values for a parameter. It helps you to specify multiple values for the parameter.



Available Values

Lists all predefined parameter values for selection. When the parameter is bound with a column, but the Display Column is different from the Bind Column, values of the Display Column are listed here.

Selected Values

Lists the values selected. The selected values are case sensitive.

Adds the selected values from the Available Values box to the Selected Values box.

Removes the selected values from the Selected Values box.

Adds all the values from the Available Values box to the Selected Values box.

Removes all the values from the Selected Values box.

Enter Values

This option is available when the parameter's Allow Type-in of Value property is set to true in JReport Designer.

Enter a value manually in the text box and then click the button next to add the value to the Selected Values box. When the parameter is bound with a column, but the display column is different from the bind column, make sure the value you enter is that of the bind column.

Opens the Calendar dialog to specify a date and time value. Available only for Date, DateTime, or Time parameter which allows for type-in values and is not bound with any column.

Search

This option is available when the parameter's Allow Type-in of Value property is set to false in JReport Designer. It is used to search for values among the available values. After you type text in the Search text box, the matched text in the available values will be highlighted.

All

If selected, it means that the parameter value result is all the values in the DBMS. This option is available when the parameter's Enable the "All" Option property is set to true in JReport Designer. This is translated in SQL to remove the parameter which may select more values than listed in the available values list.

For example, when the parameter query is:

```
SELECT CUSTOMERS.CUSTOMERID,CUSTOMERS.CUSTOMERNAME FROM CUSTOMERS where CUSTOMERS.
CUSTOMERID>0 and CUSTOMERS.CUSTOMERID<4
```

When you click >> to add 1, 2 and 3 as the parameter values, which are all the available values, the SQL is:

```
(CUSTOMERS.CUSTOMERID IN ( 1,2,3))
```

Run the same dashboard again and this time check the All checkbox in the Enter Values dialog, the query is then:

```
( 1 = 1)
```

In this case, you will get more customers even though available values are only 1 •3.

Note: When a multi-value parameter is inserted as a field into a report and All is selected as the value, the field will show the string "All". If the parameter allows type-in values, "All" means all possible values of the parameter data type; if the parameter does not, "All" represents all the values of the parameter that come from the DBMS.

OK

Closes this dialog and applies the specified values to the parameter.

Cancel

Cancels changing the parameter values and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Related topics:

- Specifying parameter values

Export dialog

There is a component level Export dialog which exports a specific library component, and a dashboard level Export dialog which allows you to export one or more library components in a dashboard into one format at a time.

Exportable library components include library components that were created using JReport Designer and then inserted into dashboards via the Resources panel.

OK

Closes this dialog and starts exporting the specified library components.

Cancel

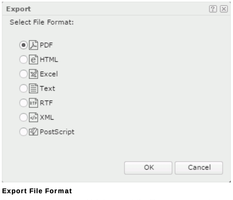
Cancels the export and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Export dialog for component level

The component level Export dialog appears after you click and select Export on the title bar of a library component. It allows you to export individual library component.

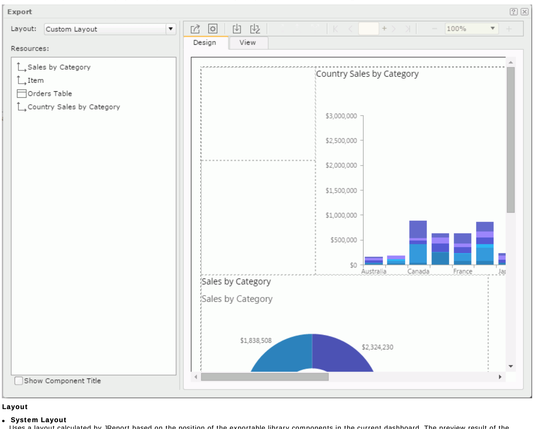


Export File Format

Specifies the format in which to export the library component.

Export dialog for dashboard level

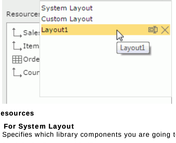
The dashboard level Export dialog appears after you click the Export button on the side bar, or click the Options button on the side bar and select Export from the option list. It helps you to export one or more library components in a dashboard using the same format.



Layout

- **System Layout**
  Uses a layout calculated by JReport based on the position of the exportable library components in the current dashboard. The preview result of the system layout is displayed on the right.

  The system layout allows you to decide the components to export and their exporting order using the Resources box.
- **Custom Layout**
  If selected, you can rearrange the layout of the library components you would like to export using a tabular. You can customize the layout in the Design tab and preview it in the Preview tab.
- **Other customized layouts saved**
  Your saved layouts are also listed for choosing. You can do further modifications based on them. For each of the customized layout, you can edit its name or delete it using the two buttons - Rename and Delete - appearing on the right when the mouse hovers over the layout item on the drop-down list.



Resources

- **For System Layout**
  Specifies which library components you are going to export and their exporting order.

  By default all exportable library components are listed and selected. To export some of them, unselect the others.

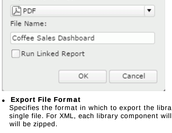
  The order of the listed library components determines the order in which they will be exported. Click the buttons and to adjust the exporting order if necessary.
- **For Custom Layout**
  Lists all the exportable library components in the dashboard. You can drag the components from the box to the Design tab to customize the layout.

Show Component Title

Specifies whether to show the library component titles in the exported result.

Export

Opens a dialog for doing the last setting and then exporting.



- **Export File Format**
  Specifies the format in which to export the library components. For a format other than XML, all selected library components will be exported into one single file. For XML, each library component will be exported to a separate XML file. When the exported result contains more than one file, all the files will be zipped.
- **File Name**
  The exported result may be either a single file or a zipped package of multiple files, so this name is either a file name or a zipped package name.
- **Run Linked Report**
  If the library components are linked with other reports, you can check whether or not to generate the linked reports in the exported result file. If you are not interested in the linked report, leave this option unchecked. Generating linked reports at the same time, especially when the linked reports contain a large amount of data, will cause performance issue.
- **OK**
  Starts exporting.
- **Cancel**
  Ignores the setting and exits the dialog.
- Displays the help document about this feature.
- Ignores the setting and closes this dialog.

Page Setup

Opens the Page Setup dialog to specify the page properties of the exported result file.

Save

Saves changes to an existing layout or opens the Save As dialog for saving a newly created layout. The saved layouts are available in the Layout drop-down list for applying. The layouts that you save will be available to all users using the same dashboard.

Save As

Opens the Save As dialog for saving a new layout with a name or saving an existing layout with another name.

Options for managing tabular cell

Available to non-system layouts. They are used for customizing the layout in the Design tab.

- **Horizontal Split**
  Splits a cell into two horizontal cells. When there is a library component in the cell, it will be placed in the left cell after the splitting.
- **Merge**
  Merges adjacent cells forming a rectangle into one cell. When there are more than one library component in the cells, the cells cannot be merged.
- **Vertical Split**
  Splits a cell into two vertical cells. When there is a library component in the cell, it will be placed in the top cell after the splitting.

Options for navigating pages

- **First**
  Goes to the start page.
- **Previous**
  Goes to the previous page.
- 1 **Turn to Page**
  Input a number in the text box and then press the Enter button on the keyboard to go to that page directly. If the input number is larger than the total page number, it goes to the last page, and if the number smaller than 1, it goes to the first page.
- **Next**
  Goes to the next page.
- **Last**
  Goes to the end page.

Options for zooming in / out the pages

100% The usage of the tool:

- Click - to zoom out.
- Click + to zoom in.
- Select a percentage from the drop-down list.
- Input a percentage in the text box. The zoom range is from 30% to 400%. If you input a value out of the range, the ratio will be restored to the previously applied percentage.

Design tab

Available to non-system layouts. It is where you customize the component layout. The page size is determined by the page setup properties and the ratio setting. A tabular fulfills the page printable area for holding a library component in each cell. Split or merge the cells and then drag library components from the Resources box into the cells to customize the layout.

When exporting, for tables and crosstabs, only their current view of data as displayed in the dashboard will be exported by default. For example, if a table contains 10 pages of data and you browse to page 3 when you open the Export dialog, then in the exported result, the table will only contain data of page 3. If you want their full data to be exported, you need to use the Filter option on the context menu to switch from Current View to All.

There are these options on the context menu after right-clicking in a tabular cell:

- **Filter**
  Opens the Filter dialog to choose the data scope of the table or crosstab in the tabular cell.
- **Remove**
  Removes the library component from the tabular cell.

View tab

Available to non-system layouts. It provides a preview of the customized layout. You can click the page navigation and zooming buttons to preview the result.

Displays the help document about this feature.

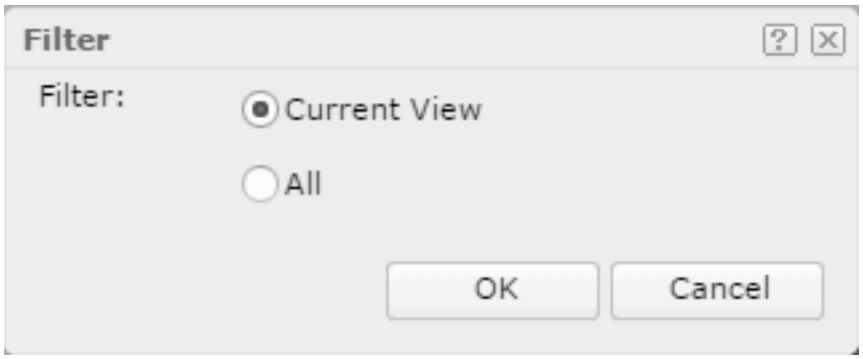
Ignores the setting and closes this dialog.

Related topics:

- Exporting library components

Filter dialog

The dialog appears when you right-click a table or crosstab in the Design tab of the Export dialog or Print dialog and choose Filter from the shortcut menu. It helps you to choose whether to export/print all the data of the library component or just the current page displayed in the dashboard.



Current View

Specifies to export/print the table or crosstab's current page displayed in the dashboard.

All

Specifies to export/print all the data of the table or crosstab.

OK

Closes this dialog and applies the setting.

Cancel

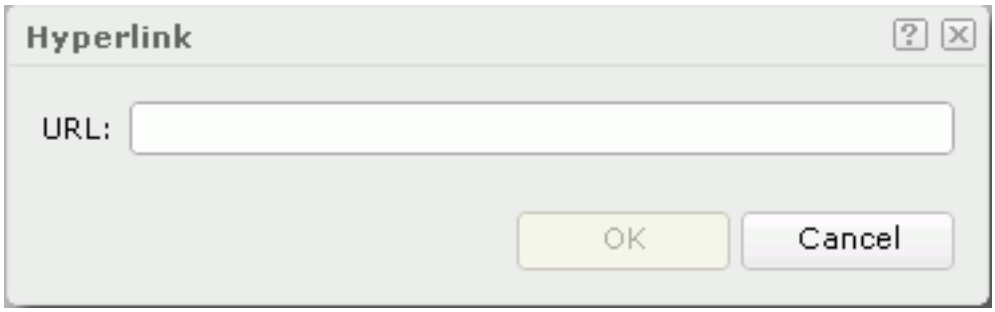
Cancels the setting and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Hyperlink dialog

This dialog appears when you click the Hyperlink button in the Edit HTML dialog or Insert HTML dialog. It helps you to insert a hyperlink on the selected object.



URL

Specifies a URL that the hyperlink will be directed to.

OK

Closes this dialog and inserts the hyperlink.

Cancel

Cancels the insertion and closes this dialog.

Displays the help document about this feature.

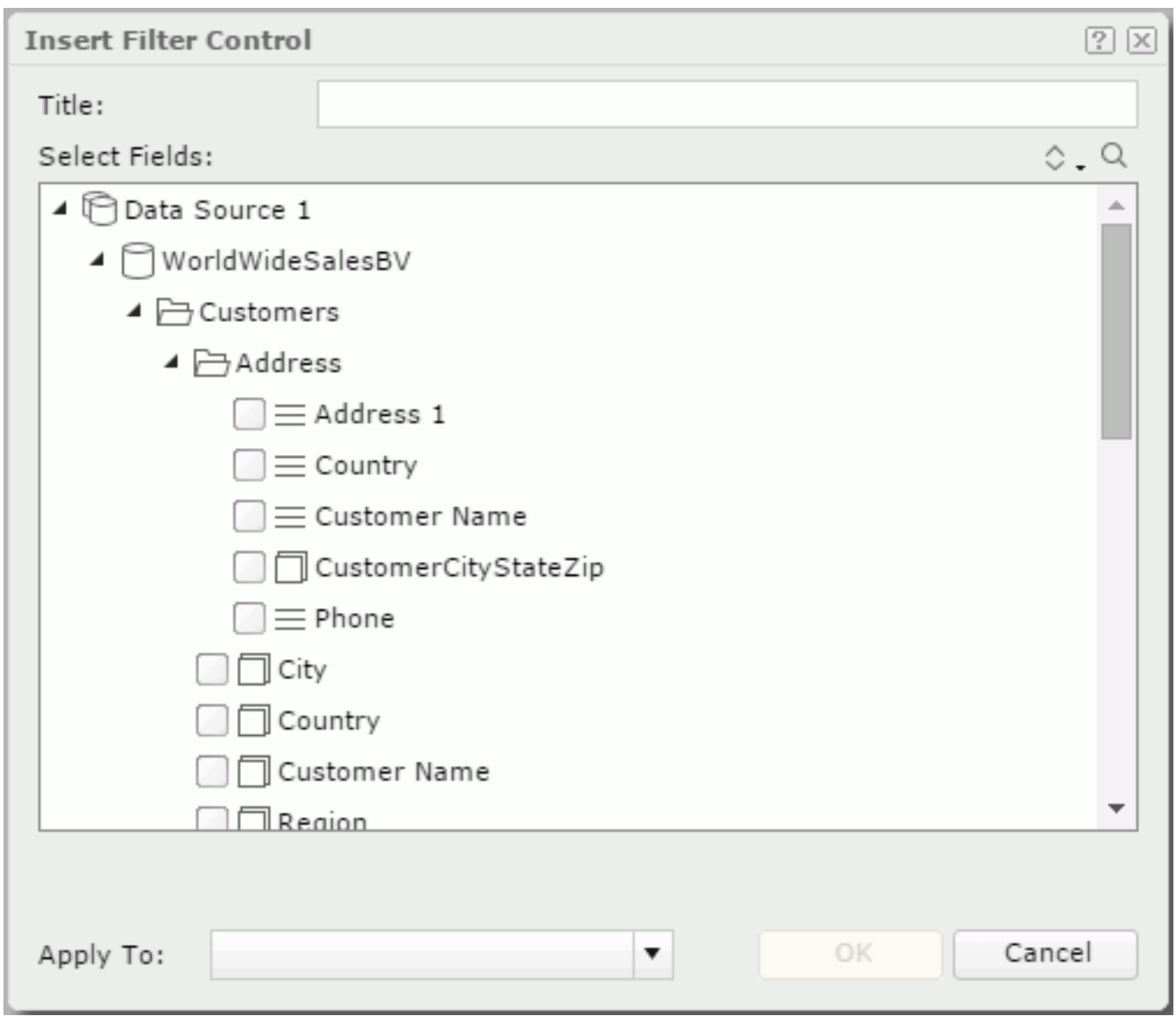
Ignores the setting and closes this dialog.

Related topics:

- Inserting an HTML component

Insert Filter Control dialog

This dialog appears when you drag Filter Control from Toolbox in the Resources panel to the dashboard body. It helps you to insert a filter control into the dashboard body to filter component data.



Title

Specifies a title for the filter control.

Select Fields

Specifies the fields to bind to the filter control. All the selected fields should be of the same data type. All uncomparable data type fields cannot be bound to filter controls, such as Binary, Blob, Clob, Longvarchar, Longvarbinary, and Varbinary.

The common usage is to select one field, and then based on the field to filter the data of the components created from the same data source as the field.

To filter components using different data sources, choose a common field all the data sources contain and select the field in all the data sources.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.
-
  Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
-
  Highlights the previous matched text.
-
  Highlights the next matched text.

Apply To

Specifies the data components from the drop-down list to apply the filter control to. <All> means all data components involving the selected fields in the dashboard.

OK

Closes this dialog and inserts the filter control in the dashboard body.

Cancel

Cancels the insertion and closes this dialog.

Displays the help document about this feature.

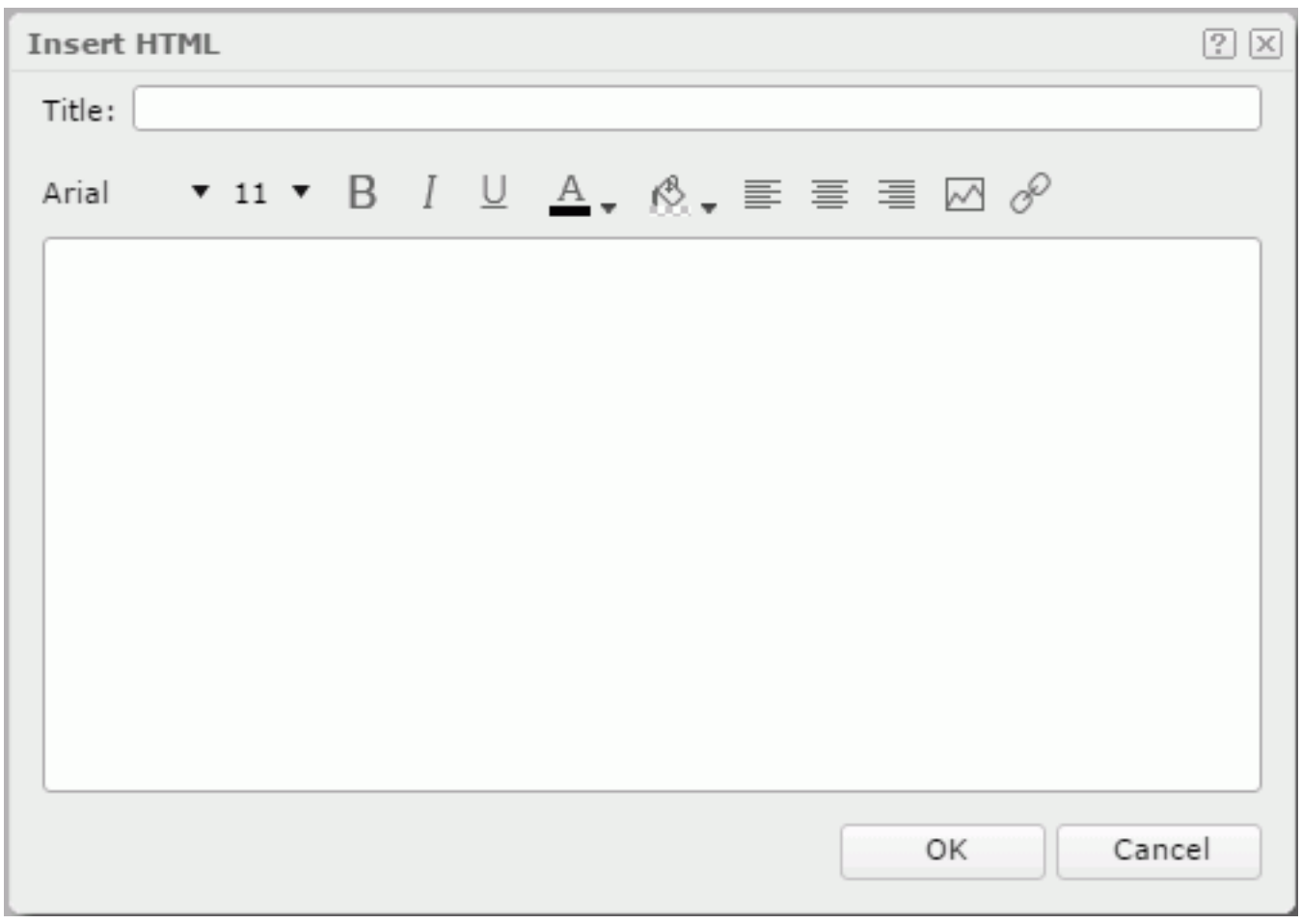
Ignores the setting and closes this dialog.

Related topics:

- Filtering component data

Insert HTML dialog

This dialog appears when you drag HTML from Toolbox in the Resources panel to the dashboard body. It helps you to insert an HTML component into the dashboard body.



Title

Specifies a title for the HTML component.

HTML text box

Type text in the box directly. You can make use of the buttons above the text box to format the text, insert images, and create hyperlinks.

Here are the buttons:

- Arial **Font Face**
  Specifies the font of the text from the drop-down list.
- 11 **Font Size**
  Specifies the font size of the text from the drop-down list.
- B
  Specifies whether to make the text bold.
- I
  Specifies whether to make the text italic.
- U
  Specifies whether to underline the text.
- A **Font Color**
  Specifies the color of the text.

  To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range.
- **Background Color**
  Specifies the background color of the text.

  To change the color, click the color indicator to select a color, or click **More Colors** in the color indicator to access the Color Picker dialog in which you can select a color within a wider range.
- **Align Left**
  Specifies to align the text to the left.
- **Align Center**
  Specifies to align the text in the center.
- **Align Right**
  Specifies to align the text to the right.
- **Insert Image**
  Opens the Insert Image dialog for inserting an image into the HTML component.
- **Hyperlink**
  Opens the Hyperlink dialog to create a hyperlink on the selected object.

OK

Closes this dialog and inserts the HTML component in the dashboard body.

Cancel

Cancels the insertion and closes this dialog.

?

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Related topics:

- Inserting an HTML component

Insert Image dialog

This dialog appears when you drag Image from Toolbox in the Resources panel to the dashboard header, or click the Insert Image button in the Insert HTML dialog or Edit HTML dialog. It helps you to insert an image into the dashboard header or into the HTML component.

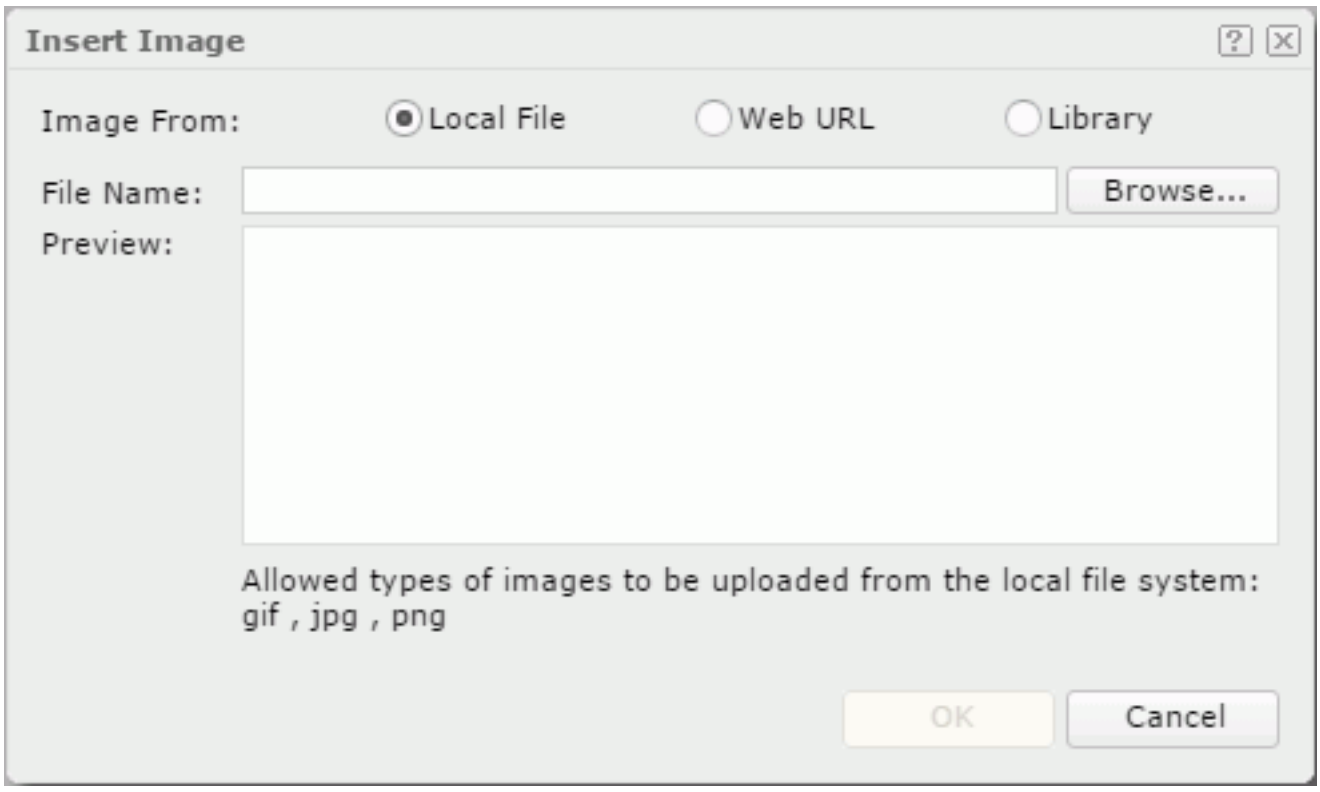


Image From

Specifies the source of the image file.

- **Local File**
  Specifies to use an image from the local file system.
  - **File Name**
    Specifies the path and name of the image file. You can click the Browse button to locate the image file.
- **Web URL**
  Specifies to use an image via URL.
  - **File URL**
    Specifies the URL of the image file.

    Note: If your JReport Server is in an intranet which requires a proxy, to successfully access the image via URL, you need to add the parameters `-Dhttp.proxyHost=XXX -Dhttp.proxyPort=XX` to the server's startup file JRServer.bat, which is located in `<install_root>\bin`.
- **Library**
  Specifies to use an existing image.
  - **My Pictures**
    The My Pictures folder is a virtual location where JReport Server stores the images that have once been inserted into JDashboard. Select the one you want to use.

Preview

Displays a preview of the selected image.

OK

Closes this dialog and inserts the selected image in the dashboard header.

Cancel

Cancels the insertion and closes this dialog.

Displays the help document about this feature.

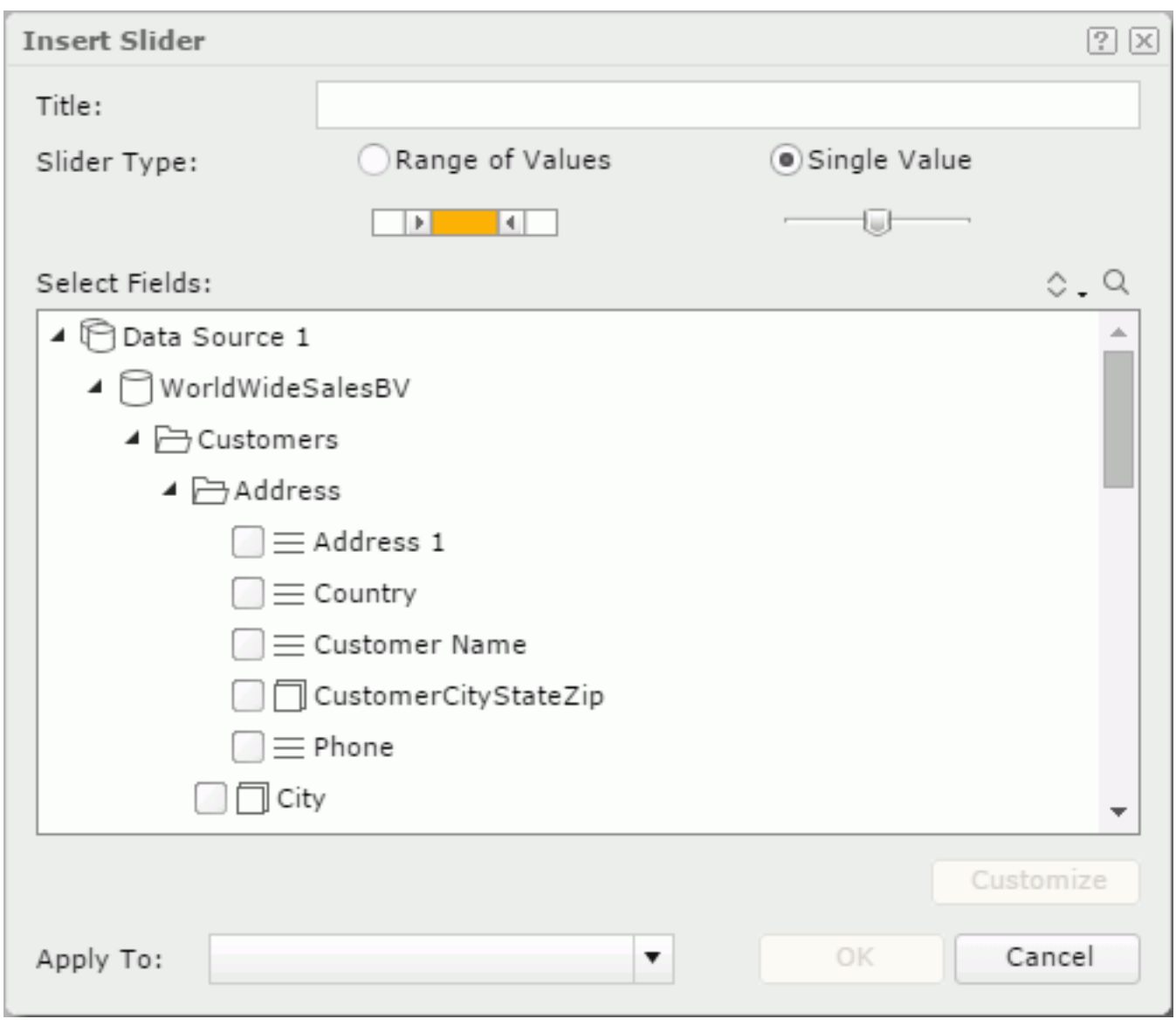
Ignores the setting and closes this dialog.

Related topics:

- Inserting an image

Insert Slider dialog

This dialog appears when you drag Slider from Toolbox in the Resources panel to the dashboard body. It helps you to insert a slider into the dashboard body to filter component data.



Title

Specifies a title for the slider.

Slider Type

Specifies whether the slider is used to specify a range of values or to specify a single value:

- **Range of Values**
  If selected, you can select a range of values at runtime.
- **Single Value**
  If selected, you can select one value at runtime. Not available for numeric data type fields.

Select Fields

Specifies the fields to bind to the slider. All the selected fields should be of the same data type.

The common usage is to select one field, and then based on the field to filter the data of the components created from the same data source as the field.

To filter components using different data sources, choose a common field all the data sources contain and select the field in all the data sources.

Sorts the view elements in the specified order from the drop-down list. Once a user changes the order, it will be applied to all the resource trees where business view elements are listed for this user.

- **Predefined Order**
  Sorts the view elements in the order defined in the Business View Editor on JReport Designer.
- **Resource Types**
  Sorts the view elements by resource type, namely category objects come first, then group objects, then aggregation objects, and at last detail objects.
- **Alphabetical Order**
  Sorts the view elements in alphabetical order. Elements that are not in any category will be sorted first, then the categories, and the elements in each category will also be sorted alphabetically.

Launches the quick search toolbar to search for view elements.

- **Text box**
  The quick search toolbar treats view elements as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the view elements.
- **X**
  Closes the quick search toolbar.
-
  Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
-
  Highlights the previous matched text.
-
  Highlights the next matched text.

Customize

Opens the Customize Value dialog for you to customize the values of the selected field.

Apply To

Specifies the data components from the drop-down list to apply the slider to. <All> means all data components involving the selected fields in the dashboard.

OK

Closes this dialog and inserts the slider in the dashboard body.

Cancel

Cancels the insertion and closes this dialog.

Displays the help document about this feature.

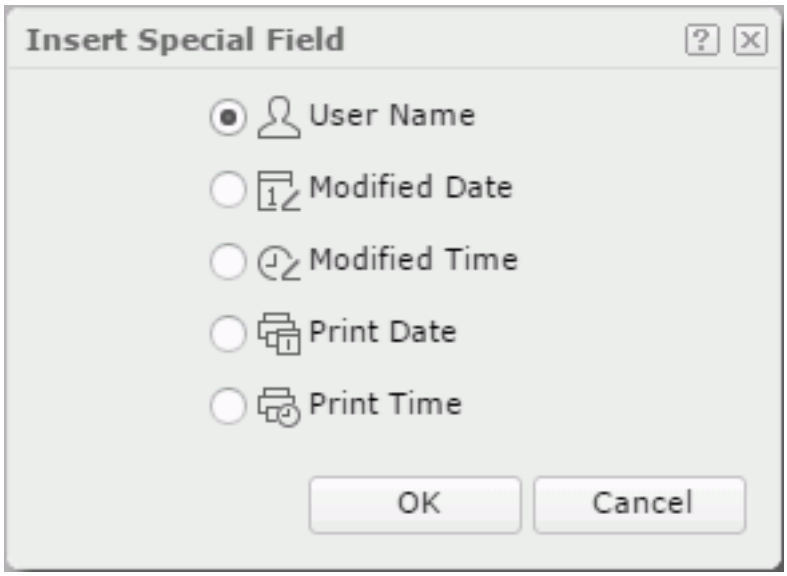
Ignores the setting and closes this dialog.

Related topics:

- Filtering component data

Insert Special Field dialog

This dialog appears when you drag Special Field from Toolbox in the Resources panel to the dashboard header. It helps you to insert a special field in the dashboard header.



- **User Name**
  The user name with which you log onto JReport Server.
- **Modified Date**
  The date when the dashboard was last modified.
- **Modified Time**
  The time when the dashboard was last modified.
- **Print Date**
  The date to run the dashboard.
- **Print Time**
  The time to run the dashboard.

OK

Closes this dialog and inserts the selected special field in the dashboard header.

Cancel

Cancels the insertion and closes this dialog.

Displays the help document about this feature.

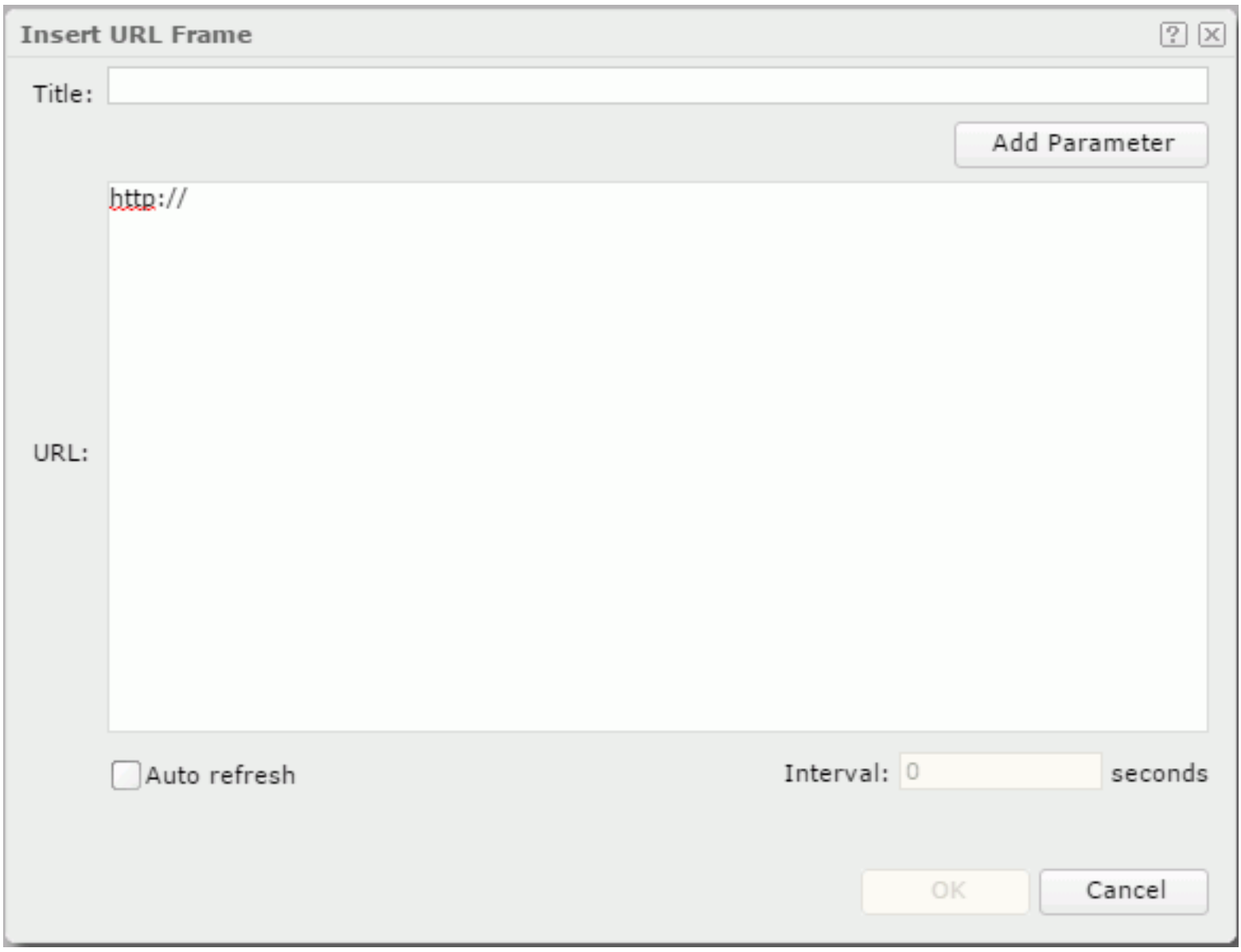
Ignores the setting and closes this dialog.

Related topics:

- Inserting a special field

Insert URL Frame dialog

This dialog appears when you drag URL Frame from Toolbox in the Resources panel to the dashboard body. It helps you to insert a web page into the dashboard.



Title

Specifies the title of the frame that holds the web page.

URL

Specifies the URL of the web page you would like to show in the dashboard.

- **Add Parameter**
  Opens the Select Parameter dialog to insert a parameter into the URL to compose a dynamic URL if a data source has already been specified, or the Select Data Source dialog to select the data source that contains the required parameter if no data source has been selected.

Auto refresh

If checked, the specified web page will be refreshed periodically.

- **Interval**
  Specifies the time interval between two refreshes.

OK

Closes this dialog and inserts the web page in the dashboard body.

Cancel

Cancels the insertion and closes this dialog.

Displays the help document about this feature.

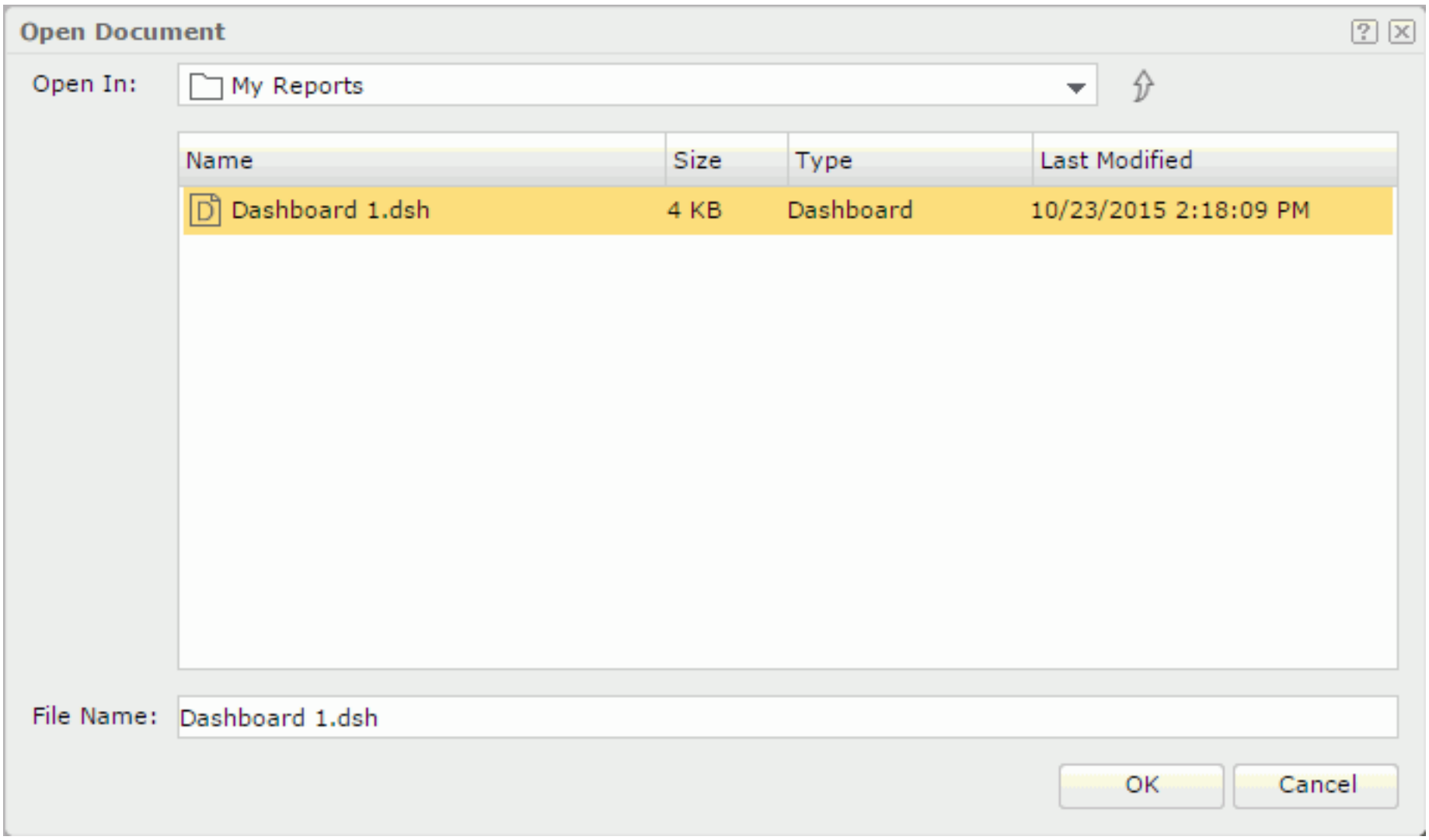
Ignores the setting and closes this dialog.

Related topics:

- Inserting a third party gadget via URL

Open Dashboard dialog

This dialog appears when you click the Open button on the side bar, or click the Options button and select Open from the option list. It helps you to open another dashboard in JDashboard.



Open In

Browse to the directory in the server resource tree where the target dashboard is located and then select the dashboard. Use the button to go to the parent folder.

The resource table shows the resources in the current directory.

File Name

Shows the selected dashboard file name.

OK

Closes this dialog and opens the selected dashboard.

Cancel

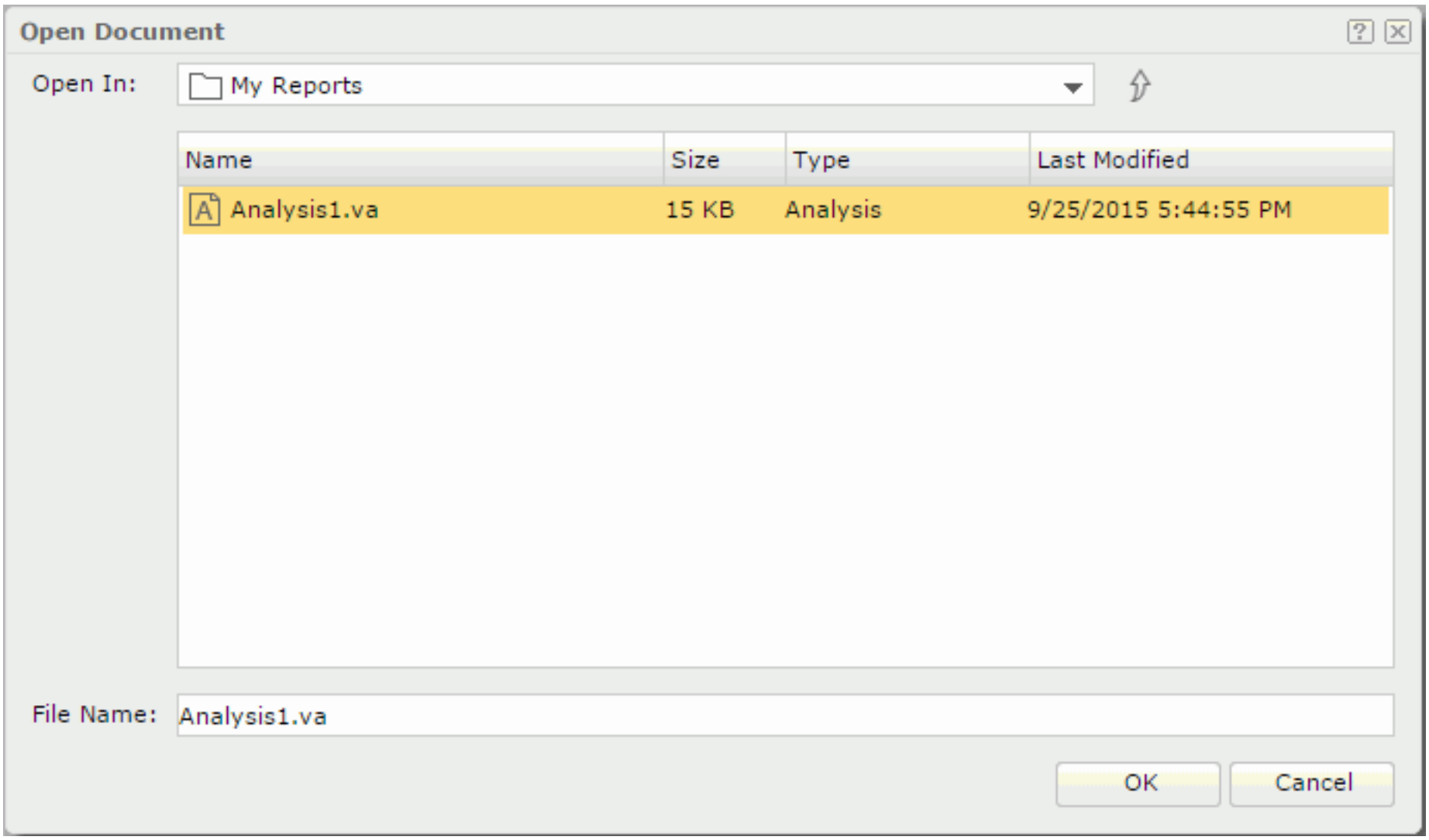
Cancels opening another dashboard and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Open Document dialog

This dialog appears when you click the Open button on the side bar, or click the Options button and select Open from the option list when your JReport Server is enabled with Visual Analysis. It helps you to open a dashboard or a Visual Analysis template in JDashboard.



Open In

Browse to the directory in the server resource tree where the target dashboard or Visual Analysis template is located and then select the resource. Use the button to go to the parent folder.

The resource table shows the resources in the current directory.

File Name

Shows the selected resource's file name.

OK

Closes this dialog and opens the selected resource.

Cancel

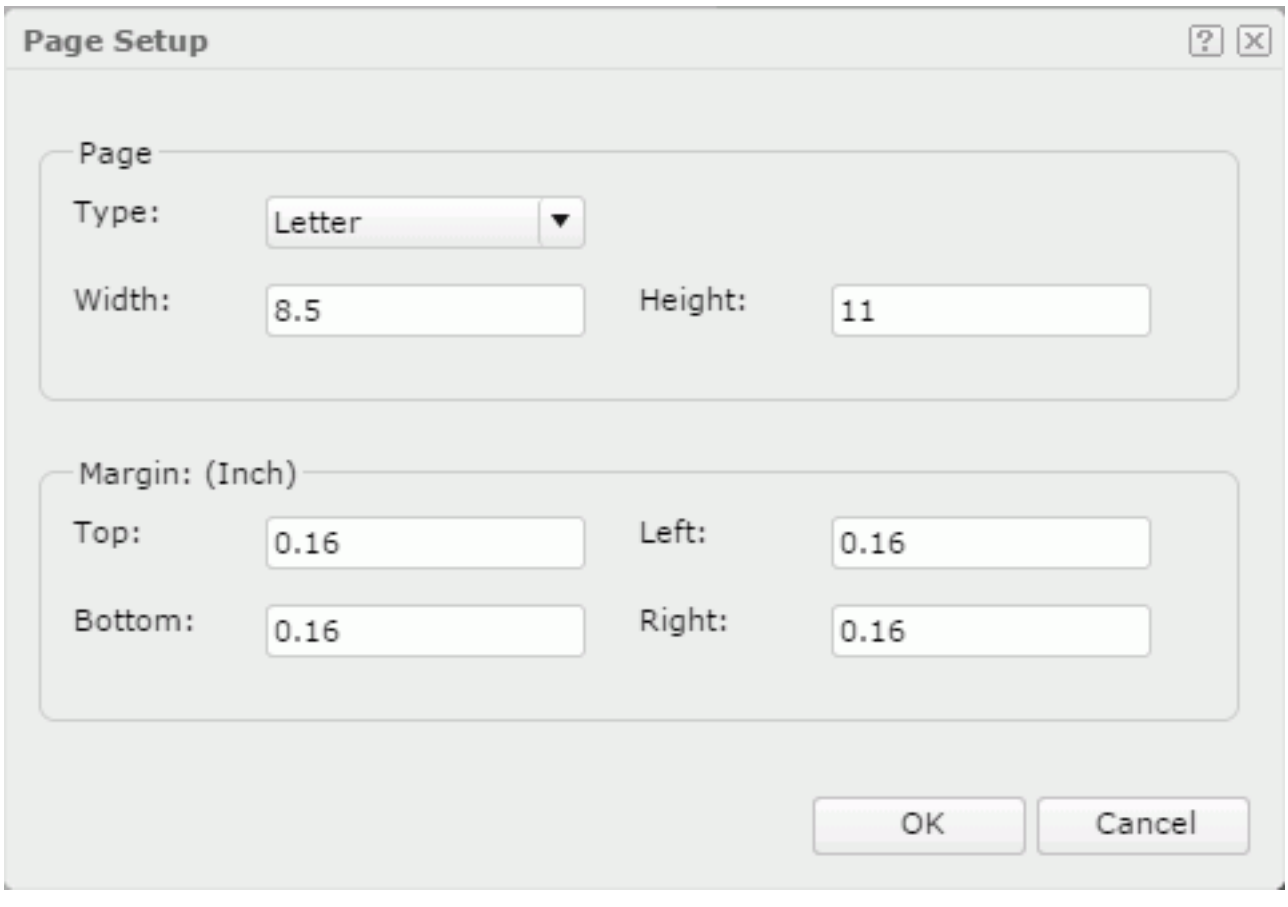
Cancels opening the resource and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Page Setup dialog

This dialog appears when you click the More Settings link in the Print dialog or click the Page Setup button in the Export dialog for dashboard level. It helps you to specify the page properties for printing or exporting a dashboard.



Page

Specifies the paper size.

- **Type**
  Specifies the paper type. If you select Custom Size, you can customize the paper's width and height. For any standard type, you can change the width and height too but it will be treated as a custom size then.
- **Width**
  Specifies the width of the paper.
- **Height**
  Specifies the height of the paper.

Margin

Specifies the margin properties.

- **Top**
  Specifies the distance of the layout area to the top edge of the page.
- **Left**
  Specifies the distance of the layout area to the left edge of the page.
- **Bottom**
  Specifies the distance of the layout area to the bottom edge of the page.
- **Right**
  Specifies the distance of the layout area to the right edge of the page.

OK

Closes this dialog and applies the setting to the printed/exported file.

Cancel

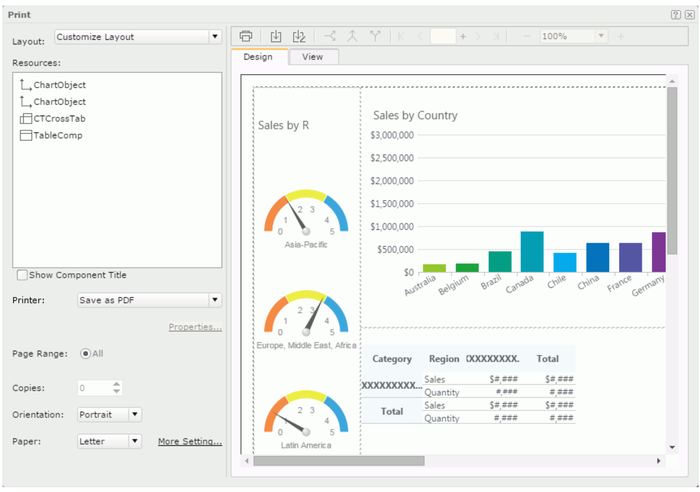
Cancels the changes and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

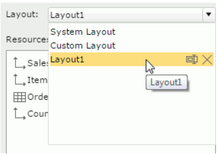
Print dialog

The dialog appears after you click the Print button on the side bar or click the Options button on the side bar and then select Print from the option list. It helps you to print one or more library components in a dashboard using the same format.



Layout

- **System Layout**
  Uses a layout calculated by JReport based on the position of the printable library components in the current dashboard. The preview result of the system layout is displayed on the right.
- **Custom Layout**
  If selected, you can rearrange the layout of the library components you would like to print using a tabular. You can customize the layout in the Design tab and preview it in the Preview tab.
- **Other customized layouts saved**
  Your saved layouts are also listed for choosing. You can do further modifications based on them. For each of the customized layouts, you can edit its name or delete it using the two buttons - Rename and Delete - appearing on the right when the mouse hovers over the layout item on the drop-down list.



Resources

- **For System Layout**
  Specifies which library components you are going to print and their printing order.

  By default all printable library components are listed and selected. To print some of them, unselect the others.

  The order of the listed library components determines the order in which they will be printed. Click the buttons and to adjust the printing order if necessary.
- **For Custom Layout**
  Lists all the printable library components in the dashboard. You can drag the components from the box to the Design tab to customize the layout.

Show Component Title

Specifies whether to show the library component titles in the print result.

Printer

Specifies the printer with which to print the library component result. In the Printer drop-down list, all the printers that the web browser can access will be listed, that is to say, the virtual printers displayed in this drop-down list vary according to the operating system and the web browser you are using. If you want to download the result to a PDF or HTML file and then use your local printer to print the file, select Save as PDF or Save as HTML.

Properties

Opens the Printer Properties dialog to specify the printing properties. Disabled when Save as PDF or Save as HTML is selected.

Page Range

Specifies the range of the pages that are to be printed.

- **All**
  If checked, all pages will be printed.

Copies

Specifies the number of copies you want to print. The number of copies will be applied to all specified pages. Disabled when Save as PDF or Save as HTML is selected.

Orientation

Specifies the orientation for the printed result.

- **Portrait**
  Prints the result in a standard letter orientation.
- **Landscape**
  Prints the result in a landscape orientation.

Paper

Specifies the type of paper for printing the result. Choose one from the drop-down list.

More Settings

Opens the Page Setup dialog to specify more page properties for the printed result.

Print

Starts printing. If Save as PDF or Save as HTML is selected in the Printer drop-down list, the result file of the library components is then opened in an associated program with which you can print the result to a printer.

Save

Saves changes to an existing layout or opens the Save As dialog for saving a newly created layout. The saved layouts are available in the Layout drop-down list for applying. The layouts that you save will be available to all users using the same dashboard.

Save As

Opens the Save As dialog for saving a new layout with a name or saving an existing layout with another name.

Options for managing tabular cell

Available to non-system layouts. They are used for customizing the layout in the Design tab.

- **Horizontal Split**
  Splits a cell into two horizontal cells. When there is a library component in the cell, it will be placed in the left cell after the splitting.
- **Merge**
  Merges adjacent cells forming a rectangle into one cell. When there are more than one library component in the cells, the cells cannot be merged.
- **Vertical Split**
  Splits a cell into two vertical cells. When there is a library component in the cell, it will be placed in the top cell after the splitting.

Options for navigating pages

- **First**
  Goes to the start page.
- **Previous**
  Goes to the previous page.
- 1 **Turn to Page**
  Input a number in the text box and then press the Enter button on the keyboard to go to that page directly. If the input number is larger than the total page number, it goes to the last page, and if the number smaller than 1, it goes to the first page.
- **Next**
  Goes to the next page.
- **Last**
  Goes to the end page.

Options for zooming in/out the pages

100% The usage of the tool:

- Click - to zoom out.
- Click + to zoom in.
- Select a percentage from the drop-down list.
- Input a percentage in the text box. The zoom range is from 30% to 400%. If you input a value out of the range, the ratio will be restored to the previously applied percentage.

Design tab

Available to non-system layouts. It is where you customize the component layout. The page size is determined by the page setup properties and the ratio setting. A tabular fulfills the page printable area for holding a library component in each cell. Split or merge the cells and then drag library components from the Resources box into the cells to customize the layout.

When printing, for tables and crosstabs, only their current view of data as displayed in the dashboard will be printed by default. For example, if a table contains 10 pages of data and you browse to page 3 when you open the Print dialog, then in the printed result, the table will only contain data of page 3. If you want their full data to be printed, you need to use the Filter option on the context menu to switch from Current View to All.

There are these options on the context menu after right-clicking in a tabular cell:

- **Filter**
  Opens the Filter dialog to choose the data scope of the table or crosstab in the tabular cell.
- **Remove**
  Removes the library component from the tabular cell.

View tab

Available to non-system layouts. It provides a preview of the customized layout. You can click the page navigation and zooming buttons to preview the result.

Displays the help document about this feature.

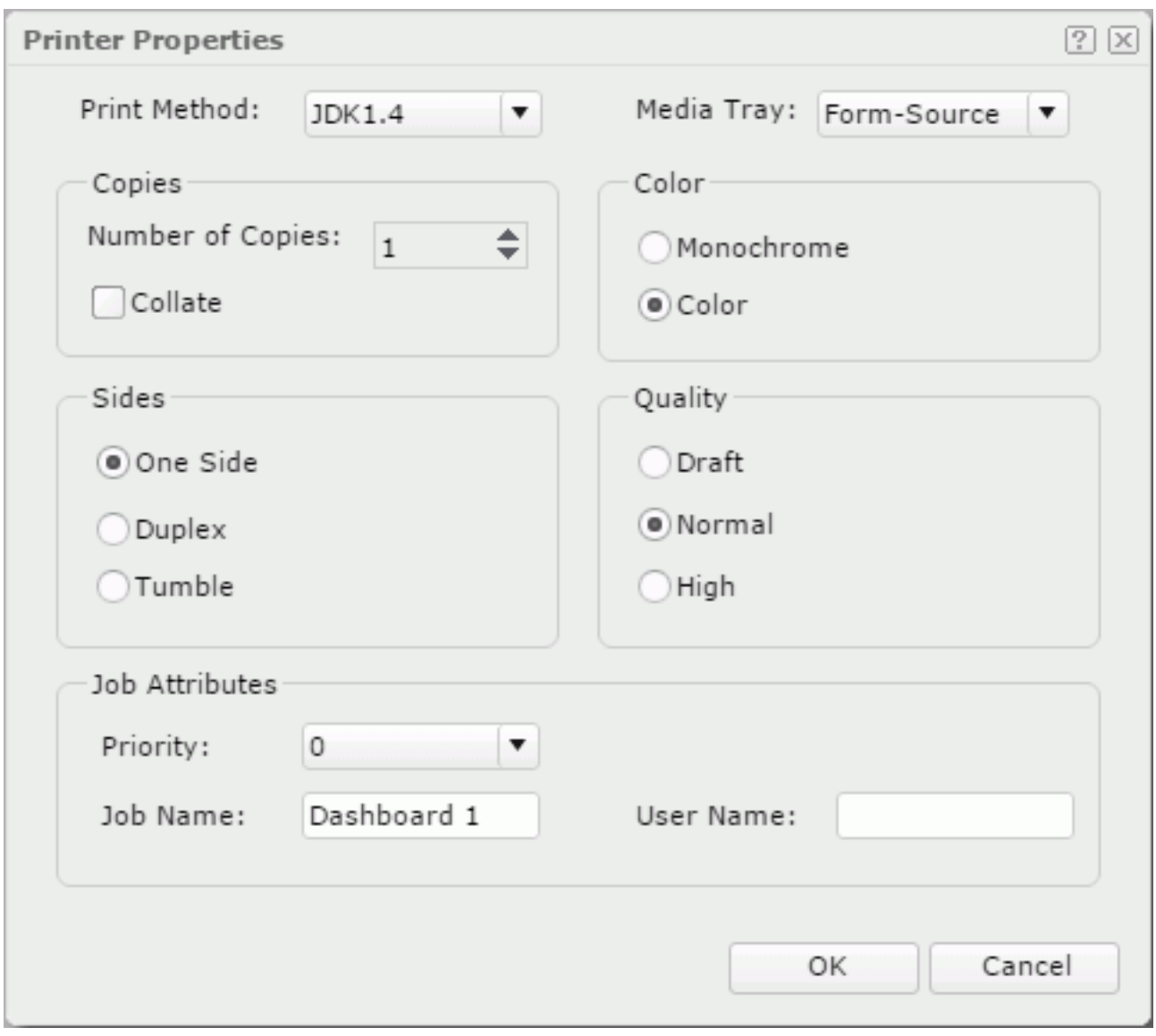
Ignores the setting and closes this dialog.

Related topics:

- Printing library components

Printer Properties dialog

This dialog appears when you click the Properties link in the Print dialog. It helps you to specify the printing properties.



Print Method

Specifies the printing method to print the result.

Media Tray

Specifies the media tray for the printer. Select one from the drop-down list.

Copies

Specifies the information of the copies.

- **Number of Copies**
  Specifies the number of copies you want to print. The number of copies will be applied to all specified pages.
- **Collate**
  Specifies the printing order for printing multiple pages. If checked, when you print two copies of a three-page result, the page order is 1, 2, 3; 1, 2, 3. Otherwise, the order will be 1, 1, 2, 2, 3, 3.

Color

Specifies the color of the result to be printed.

- **Monochrome**
  Specifies to print the result in Monochrome.
- **Color**
  Specifies to print the result in Color.

Sides

Specifies the print sides for the printed result.

- **One Side**
  Imposes each consecutive print-stream page upon the same side of consecutive media sheets.
- **Duplex**
  Imposes each consecutive pair of print-stream pages upon the front and back sides of consecutive media sheets, so that the orientation of each pair of print-stream pages on the medium is correct for the reader as if for binding on the long edge.
- **Tumble**
  Imposes each consecutive pair of print-stream pages upon the front and back sides of consecutive media sheets, so that the orientation of each pair of print-stream pages on the medium is correct for the reader as if for binding on the short edge.

Quality

Specifies the printing quality.

- **Draft**
  Specifies to print the result in draft quality.
- **Normal**
  Specifies to print the result in normal quality.
- **High**
  Specifies to print the result in high quality.

Job Attributes

Specifies the properties of the print task.

- **Priority**
  Specifies the print priority for this task.
- **Job Name**
  Specifies the name of the print task.
- **User Name**
  Specifies the name of the user.

OK

Closes this dialog and applies the setting when printing.

Cancel

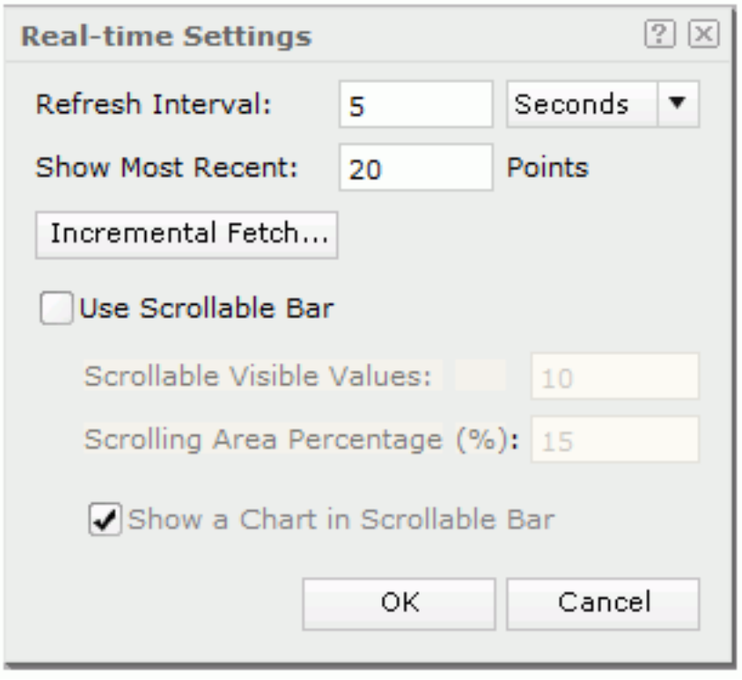
Cancels the changes and closes this dialog.

Displays this help document.

Ignores the setting and closes this dialog.

Real-time Settings dialog

The dialog appears when you right-click a real time chart and select Real-time Settings from the shortcut menu. It helps you to configure the settings for the real time chart.



Refresh Interval

Specifies the time interval at which the chart will get data and refresh itself automatically.

Show Most Recent N Points

Specifies the number of records that will be kept for the real time data on the chart.

Incremental Fetch

Opens the Unique Key dialog to configure a unique key for the real time chart.

Use Scrollable Bar

If checked, a scrollbar will be added to the chart, with which you can control the visible value range on the X axis of the chart.

- **Scrollable Visible Values**
  Specifies how many data items will be selected on the scrollbar and displayed on the axis by default.
- **Scrolling Area Percentage**
  Specifies the percentage the scrollbar occupies the whole size of the chart.
- **Show a Chart in Scrollable Bar**
  Specifies whether to show the thumbnail chart on the scrollbar.

OK

Closes this dialog and applies the settings to the real time chart.

Cancel

Cancels configuring the settings and closes this dialog.

Displays the help document about this feature.

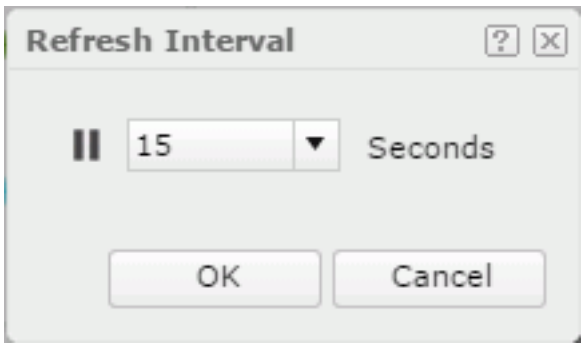
Ignores the setting and closes this dialog.

Related topics:

- Configuring real time chart settings

Refresh Interval dialog

The dialog appears when you click on the component title bar of a library component, which uses only one business view as the data source and has been enabled with the Auto Refresh feature, and select Refresh from the option list. It helps you to pause or resume the auto refresh action and specify the time interval at which to automatically refresh the library component.



/

Pauses/resumes the refresh action.

Interval

Specifies the time interval at which to automatically refresh the library component. Currently there are only 3 values available, which are ordinal multiples (starting from 2) of the default interval value defined in JReport Designer.

OK

Closes this dialog and applies the setting.

Cancel

Cancels the setting and closes this dialog.

Displays the help document about this feature.

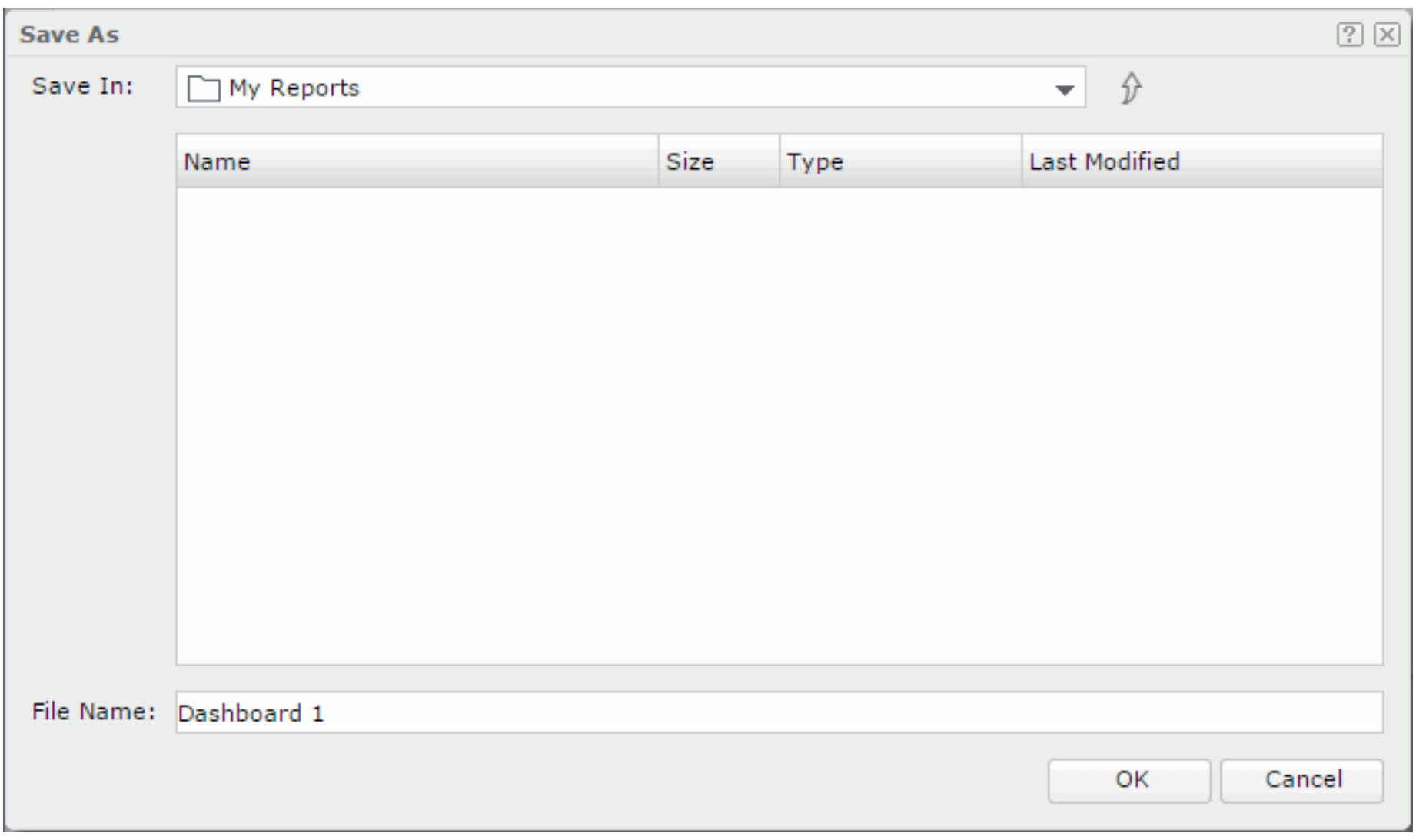
Ignores the setting and closes this dialog.

Save As dialog

This dialog is used to save different resources when it is accessed from different places.

- When accessed via the Save or Save As option on the JDashboard side bar, it is used to save the current dashboard.
- When accessed by clicking the Save or Save As button in the Export dialog or Print dialog, it is used to save a newly created export or print layout or an existing layout with a different name.

When the dialog is used to save dashboards, it contains the following options.



Save In

Specifies the directory in the server resource tree where you want to save the dashboard. Use the button to go to the parent folder. The root folder cannot be used to store resources.

Dashboards are saved in the My Reports and Public Reports folders together with reports.

The resource table shows the resources in the current directory. Click the column names to change the order of the resources in the table list if required.

- **Name**
  Displays the file names.
- **Size**
  Displays the file size.
- **Type**
  Displays the file type.
- **Last Modified**
  Displays the last modified time of the file.

File Name

Specifies the file name of the dashboard you are going to save, without suffix.

OK

Closes this dialog and saves the resource.

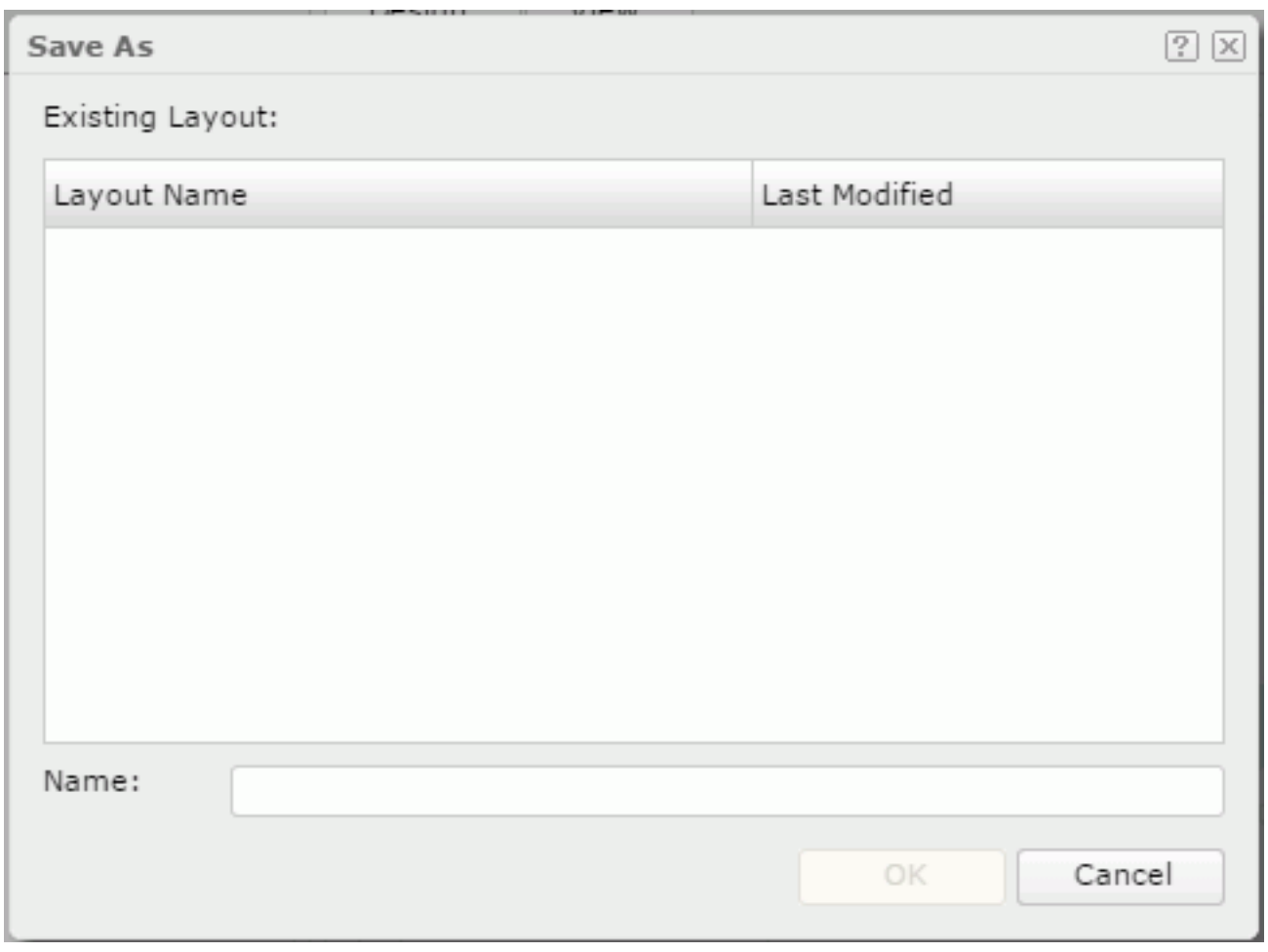
Cancel

Cancels saving the resource and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

When the dialog is for saving export or print layouts, it contains the following options.



Existing Layout

Lists the names of the already saved layouts and the time when they were last modified.

Name

Specifies the name of the layout you are going to save. Characters supported in file names can be used here. No suffix is required.

OK

Saves the current layout.

Cancel

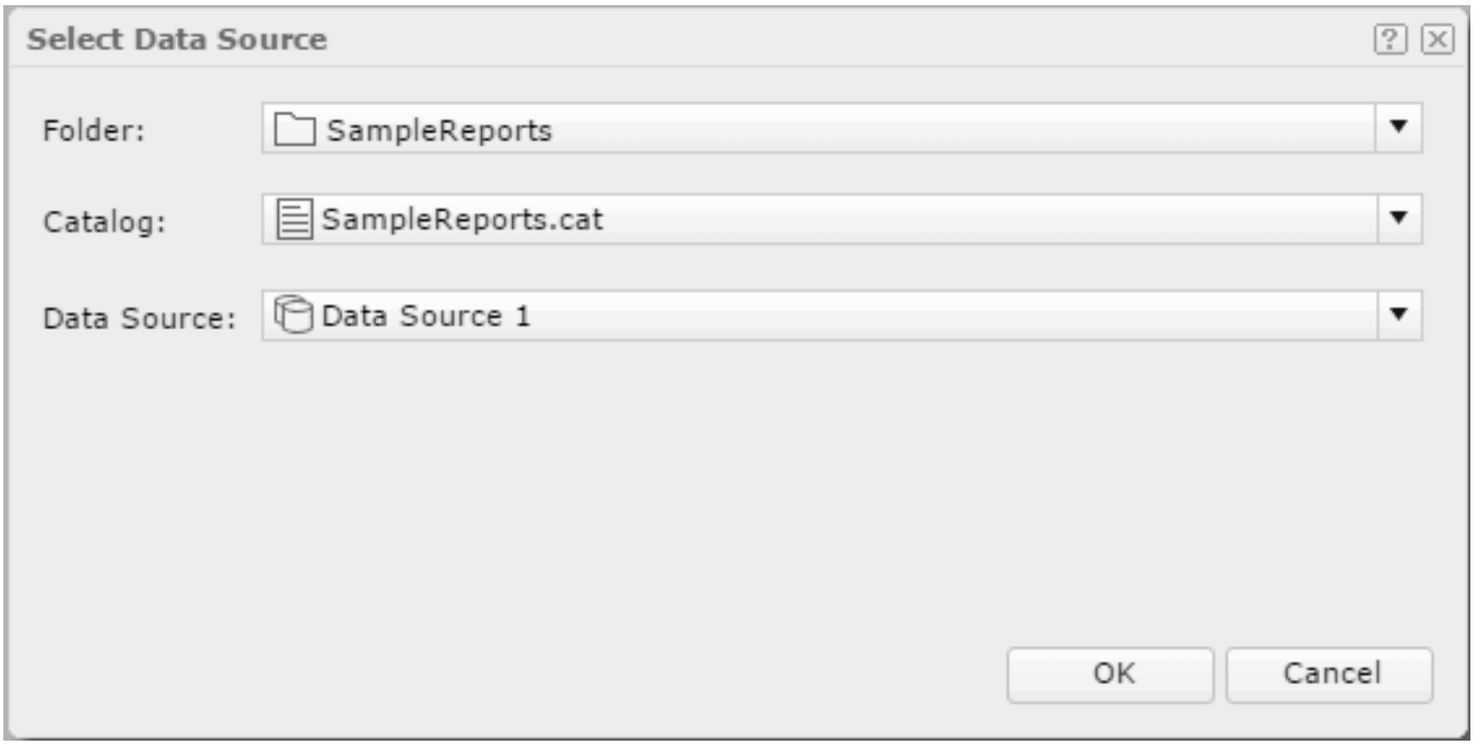
Ignores the setting and exits the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Select Data Source dialog

This dialog appears when you click the Add Parameter button in the Insert URL Frame dialog or Edit URL Frame dialog if no data source has been specified, or click the Change Data Source button in the Select Parameter dialog. It helps you to select the data source that contains the parameter you want to insert to a URL.



Folder

Specifies the folder containing the required data source.

Catalog

Displays the catalog containing the required data source.

Data Source

Displays the selected data source.

OK

Selects the data source and closes this dialog.

Cancel

Cancels the selection and closes this dialog.

Displays the help document about this feature.

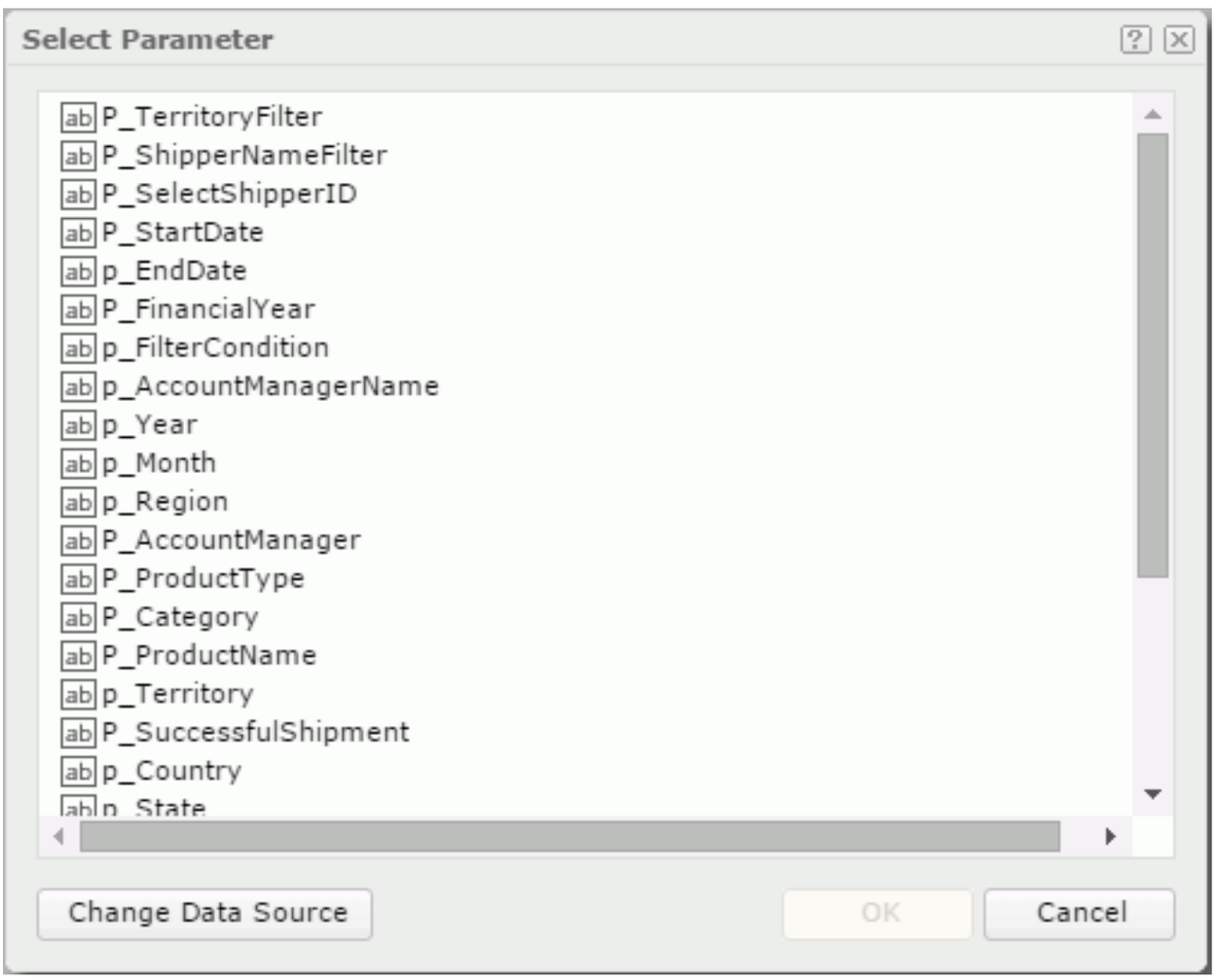
Ignores the setting and closes this dialog.

Related topics:

- Inserting a third party gadget via URL

Select Parameter dialog

This dialog appears when you click the Add Parameter button in the Insert URL Frame dialog or Edit URL Frame dialog if a data source has already been specified, or select a data source and click OK in the Select Data Source dialog. It helps you to insert a parameter into the URL of the web page that you want to add to a dashboard.



Parameters

Lists all the available parameters in the currently selected data source for you to select one to insert to the URL.

Change Data Source

Opens the Select Data Source dialog to change the data source.

OK

Closes this dialog and inserts the selected parameter to the URL.

Cancel

Cancels the insertion and closes this dialog.

Displays the help document about this feature.

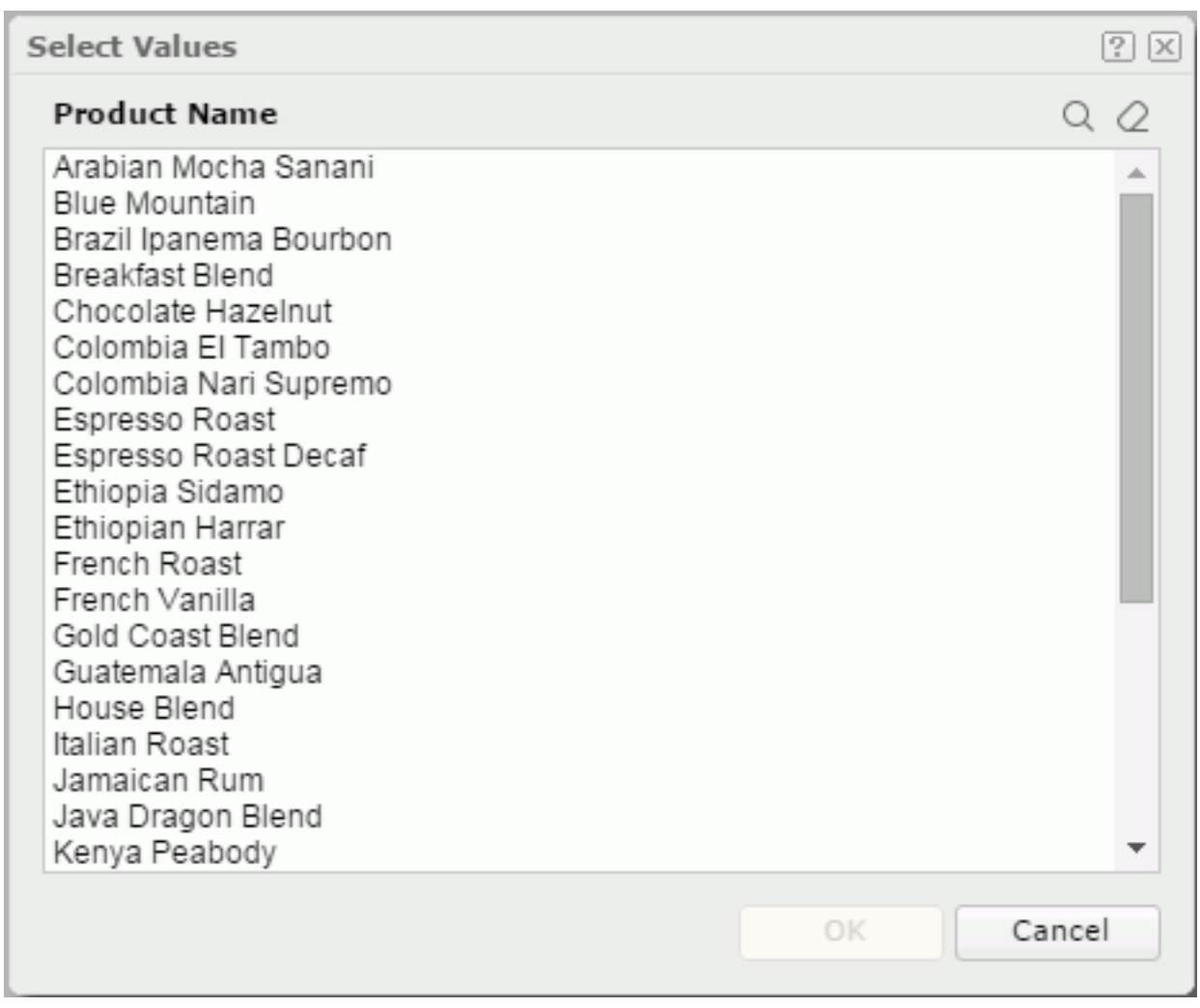
Ignores the setting and closes this dialog.

Related topics:

- Inserting a third party gadget via URL

Select Values dialog

This dialog appears when you right-click on any value of a table detail field and select Filter > More from the shortcut menu. It helps you to specify a value to apply a filter criterion.



Available values

Specifies the values to filter data with. Multiple values can be selected.

When there are more than 300 values, JReport will use Big Data Loading logic. In this case, the Shift Key for multiple selection does not work.

Search

Launches the quick search toolbar to search for values.

- **Text box**
  The quick search toolbar treats field values as strings and searches by consecutive text. Type in the text you want to search for in the text box and the matched text will be highlighted among the values.
- **X**
  Closes the quick search toolbar.
-
  Lists more search options.
  - **Highlight All**
    Specifies whether to highlight all matched text.
  - **Match Case**
    Specifies whether to search for text that meets the case of the typed text.
  - **Match Whole Word**
    Specifies whether to search for text that looks the same as the typed text.
  - **Advanced**
    Displays the quick search toolbar in the advanced mode. With the advanced options, you can customize the value range as you want.

    The advanced options available on the search toolbar vary with field types:

    - **For fields of String type**

      There are two search conditions. Select one and specify characters in the text box, then click the **OK** button on the search toolbar to start searching. To clear the characters you enter, click **X** in the text box. The Cancel button is used to close the search toolbar.

      - **Start with**
        Searches for the first characters among the values.
      - **End with**
        Searches for the last characters among the values.
    - **For fields of Numeric type**

      Five operators are supported for composing the search condition. Select one from the operator drop-down list and input value in the text box to compose the condition, then click the **OK** button on the search toolbar to start searching. To clear the value you enter, click **X** in the text box. The Cancel button is used to close the search toolbar.

      - **>**
        Greater than
      - **> =**
        Greater than or equal to
      - **<**
        Less than
      - **< =**
        Less than or equal to
      - **between**
        The data values are located between a range of values indicated by a start value and an end value. If selected, the second input value should not be smaller than the first input value, otherwise the search result will be null.
    - **For fields of Date / Time type**

      Five operators are supported for composing the search condition. Select one from the operator drop-down list and click the calendar button to select a value to compose the condition, then click the **OK** button on the search toolbar to start searching. The Cancel button is used to close the search toolbar.
-
  Highlights the previous matched text.
-
  Highlights the next matched text.

Clear

Cancels the selection of values.

OK

Closes the dialog and filters the field with the value you specified.

Cancel

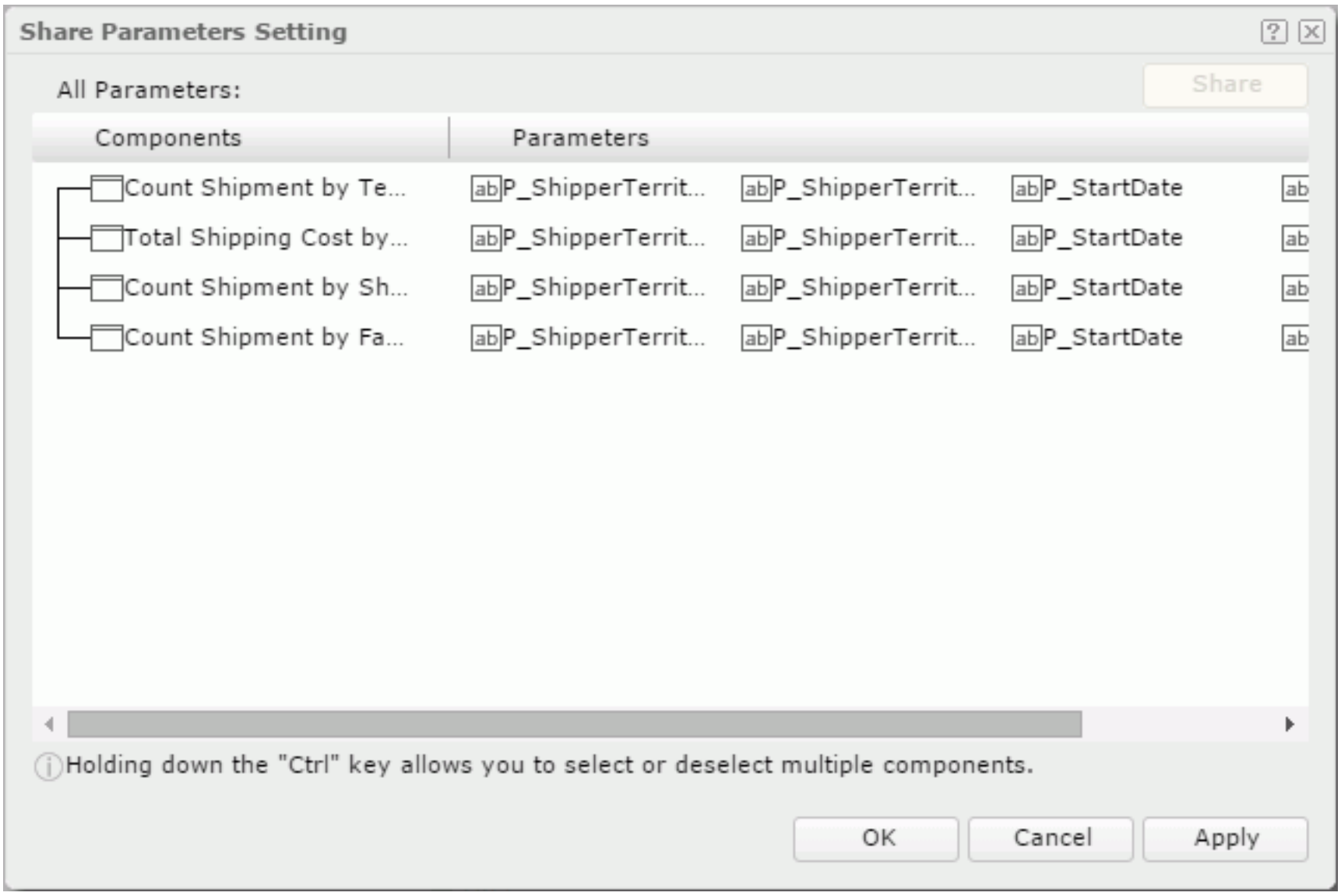
Cancels filtering the field and closes the dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Share Parameters Setting dialog

The dialog appears when you click the Options button on the side bar and then select Share Parameter from the option list. It lists all the parameters used in the current dashboard and helps you to share parameters among library components.



All Parameters

Lists all the parameters used in the library components in the current dashboard. Cascading parameters in the same group are placed within one row.

- To share parameters, hold the **Ctrl** key to select the target parameters and then click the **Share** button. The parameters will be added into one sharing group.
- To add another parameter into a sharing group, select any parameter in the group while holding Ctrl and then select the parameter, then click **Share**.
- To remove parameters from a sharing group, select the parameters and then click **Cancel Share**.

OK

Closes this dialog and applies the setting.

Cancel

Cancels the setting and closes this dialog.

Apply

Applies the setting without exiting the dialog.

Displays the help document about this feature.

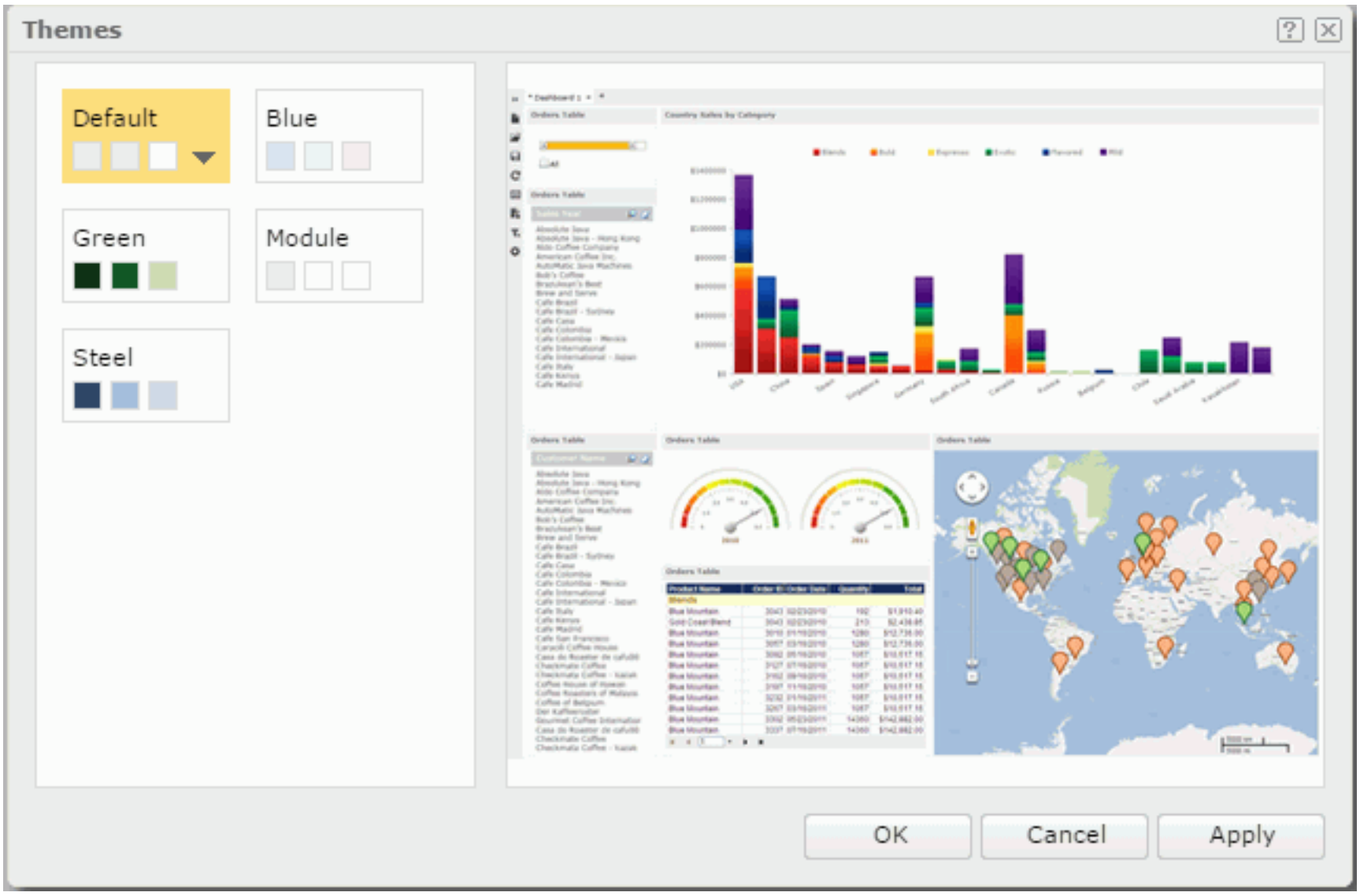
Ignores the setting and closes this dialog.

Related topic:

- Sharing parameters between library components

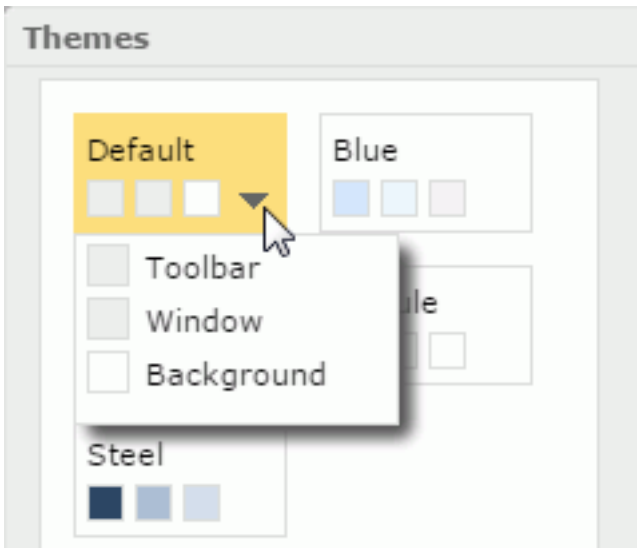
Themes dialog

This dialog appears after you click on the side bar and then select Themes from the option list. It helps you to change the theme of the current dashboard.



Theme list

The left panel lists the themes for choosing. Each theme is displayed in a rectangle with the name and three color boxes. Hover the cursor on a rectangle to show an arrow at the right most of the rectangle. Move the cursor on the arrow and when the cursor becomes a hand icon, click the arrow to show that the color in each color box is for which window element.



Preview

The right panel shows a preview of the current dashboard window with the selected theme.

OK

Closes this dialog and applies the selected theme to the current dashboard.

Cancel

Cancels changing the theme and closes this dialog.

Apply

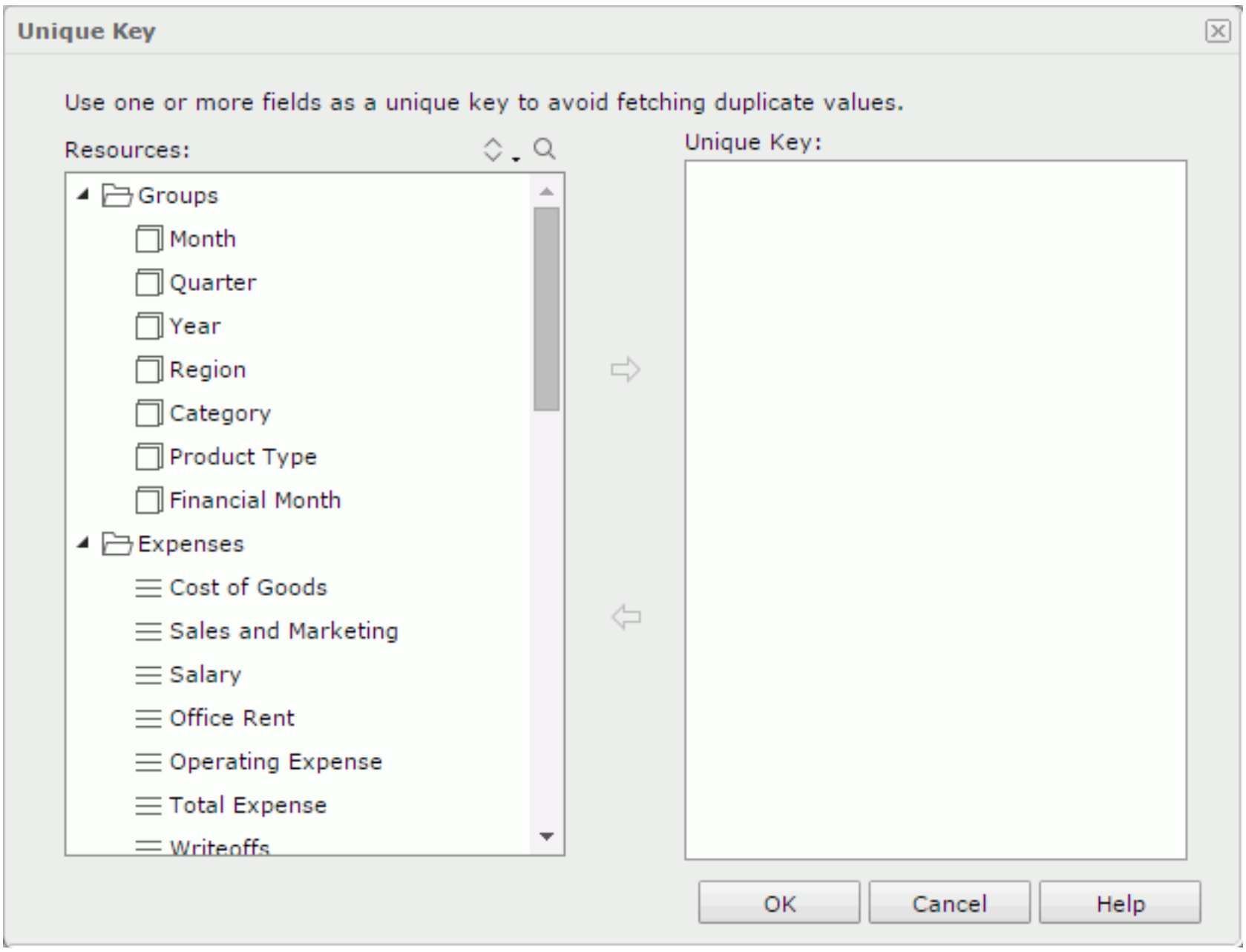
Applies the selected theme to the current dashboard without closing this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Unique Key dialog

The dialog appears when you click the Incremental Fetch button in the Real-time Settings dialog. It helps you to specify a unique key for a real time chart.



Resources

Lists all the available data resources in the dataset used by the real time chart.

Adds the selected fields to the Unique Key box.

Removes the selected fields from the Unique Key box if they are not required.

Unique Key

Lists the fields you have added as the unique key, which is used to guarantee there are no two records with the same unique key value in the real time chart.

When a real time chart is specified with a unique key, each time when the chart automatically updates itself, the data that has the same unique key value as the previous loaded data records will be filtered and only data with different unique key value will be added to the chart. For instance, if you add the fields Country and Product ID as the unique key of a real time chart, when a record with the product ID 1 in USA has already been loaded into the chart, no more records of this product ID in USA will be added to the real time chart because they have the same unique key value.

OK

Closes this dialog and applies the unique key to the real time chart.

Cancel

Cancels specifying the unique key and closes this dialog.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Related topics:

- Configuring real time chart settings

Visual Analysis dialogs

While using Visual Analysis, you will go through the following dialogs. This section introduces each of these dialogs in alphabetical order. You can go into the links for details.

- Color Picker dialog
- Edit Color dialog
- Edit Gradient Color dialog
- Edit Shapes dialog
- Open dialog
- Save As dialog
- Select Data Source dialog
- Sort dialog

Color Picker dialog

This dialog helps you to specify a color. It contains the following tabs:

- Standard tab
- Custom tab

OK

Closes this dialog and returns the color value specified.

?

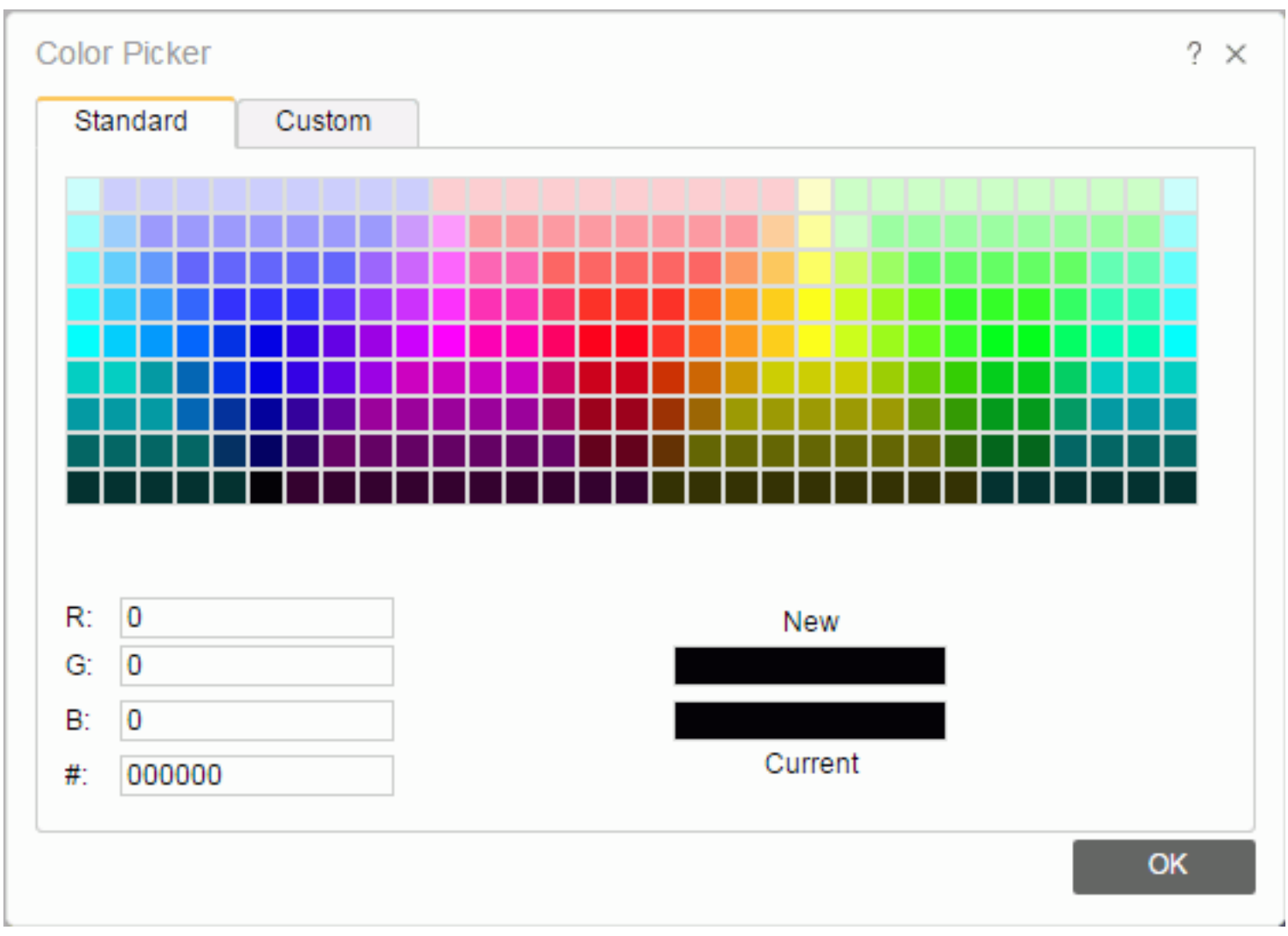
Displays the help document about this feature.

×

Ignores the setting and closes this dialog.

Standard tab

This tab allows you to select a color from the color palette.



color palette

Lists the colors for you to select one.

R

Specifies the amount of red in a color. The value is from 0 to 255.

G

Specifies the amount of green in a color. The value is from 0 to 255.

B

Specifies the amount of blue in a color. The value is from 0 to 255.

#

Specifies the hexadecimal value of the color.

New

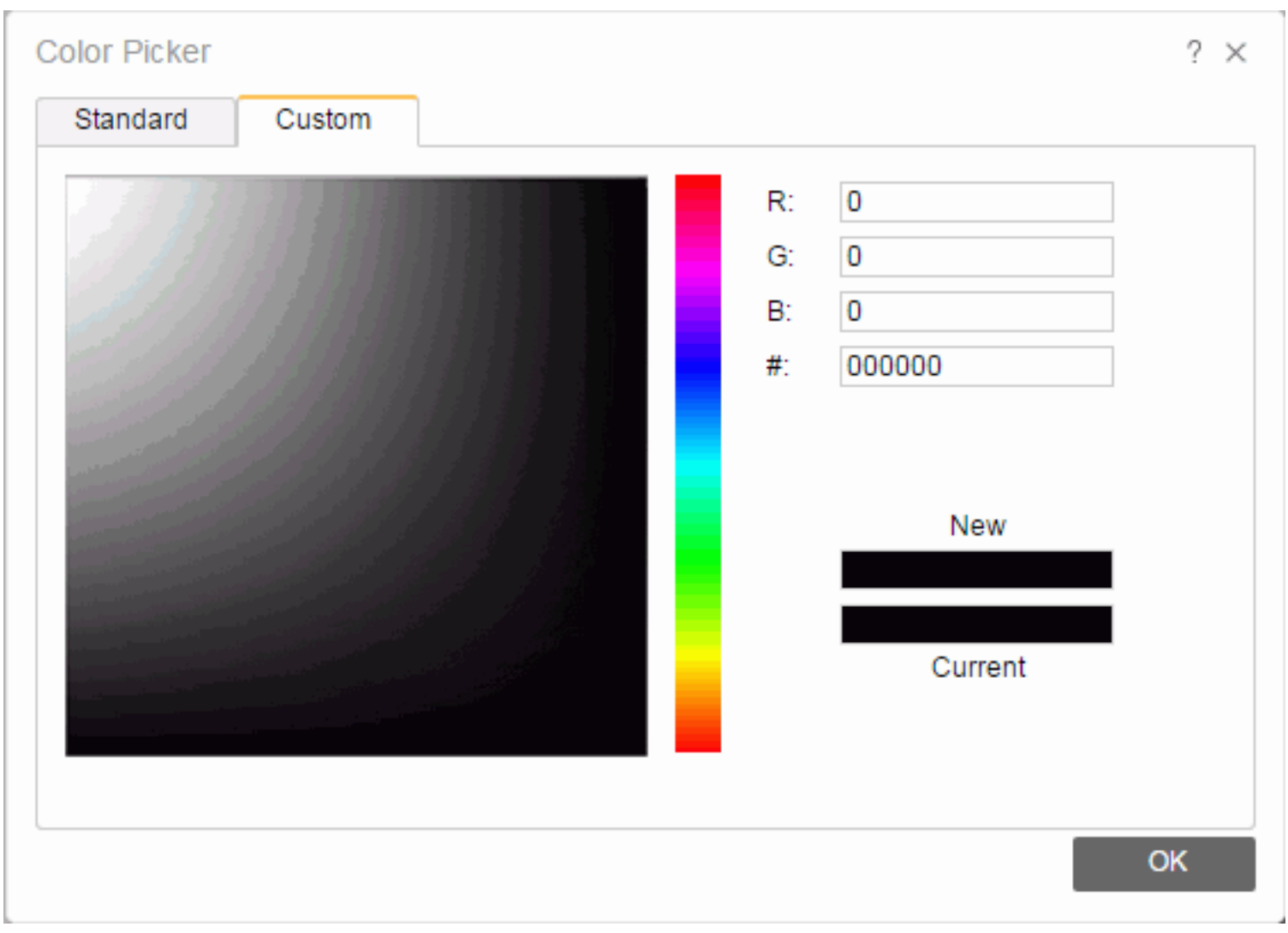
This part displays the color you select.

Current

This part always displays the original color.

Custom tab

This tab allows you to customize the color within a wider range.



Color matrix

Select a color you want in the color matrix. The hue of the matrix is based on the color you define on the color bar.

Color bar

Specifies a color on the bar which will be the hue of the color matrix.

R

Specifies the amount of red in a color. The value is from 0 to 255.

G

Specifies the amount of green in a color. The value is from 0 to 255.

B

Specifies the amount of blue in a color. The value is from 0 to 255.

#

Specifies the hexadecimal value of the color.

New

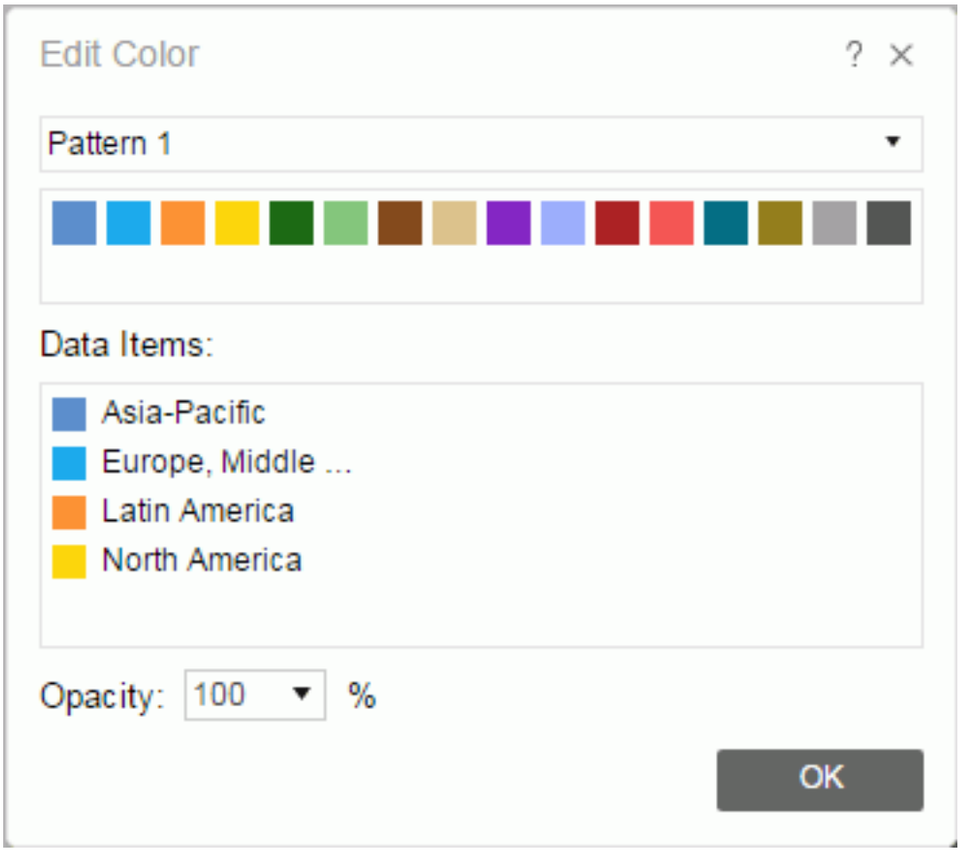
This part displays the color you select.

Current

This part always displays the original color.

Edit Color dialog

This dialog helps you to specify colors for the members of group fields.



Pattern

After selecting a pattern from the drop-down list, a preview of the colors of the pattern is provided below the pattern name.

Data Items

Lists the combination of members of the bound fields with the colors in the selected pattern.

To customize the color of a data item, take either method of the following:

- Select the data item and then select a color in the pattern preview box.
- Double-click the data item, then in the displayed color palette select a color or click **More Colors** to access the Color Picker dialog in which you can select a color within a wider range or input a color string in the format #RRGGBB.

Opacity

Specifies the opacity of the color for the selected member combination in the Data Items box.

OK

Closes this dialog and returns the specified color.

?

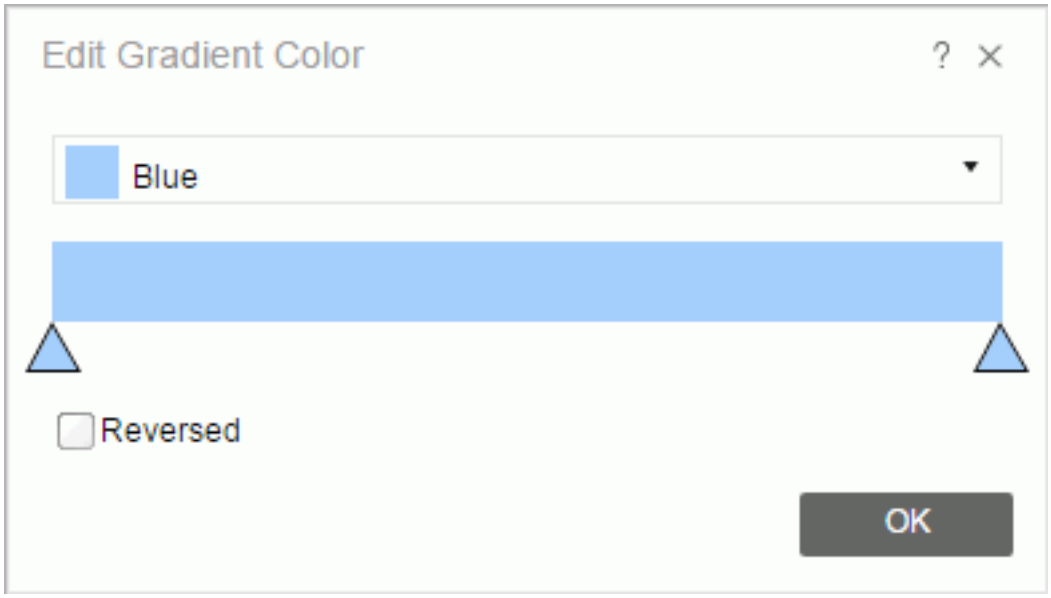
Displays the help document about this feature.

×

Ignores the setting and closes this dialog.

Edit Gradient Color dialog

This dialog helps you to specify gradient colors for the members of the aggregation field.



Pattern

After selecting a pattern from the drop-down list, a preview of the gradient color of the pattern is provided below the pattern name.

The gradient start color and end color can be customized. Click the leftmost or the rightmost arrow to select a color in the color palette or click **More Colors** to access the Color Picker dialog in which you can select a color within a wider range. After you change a color, the pattern name will be changed to Customize.

Reversed

Reverses the direction of the gradient color.

OK

Closes this dialog and returns the specified color.

?

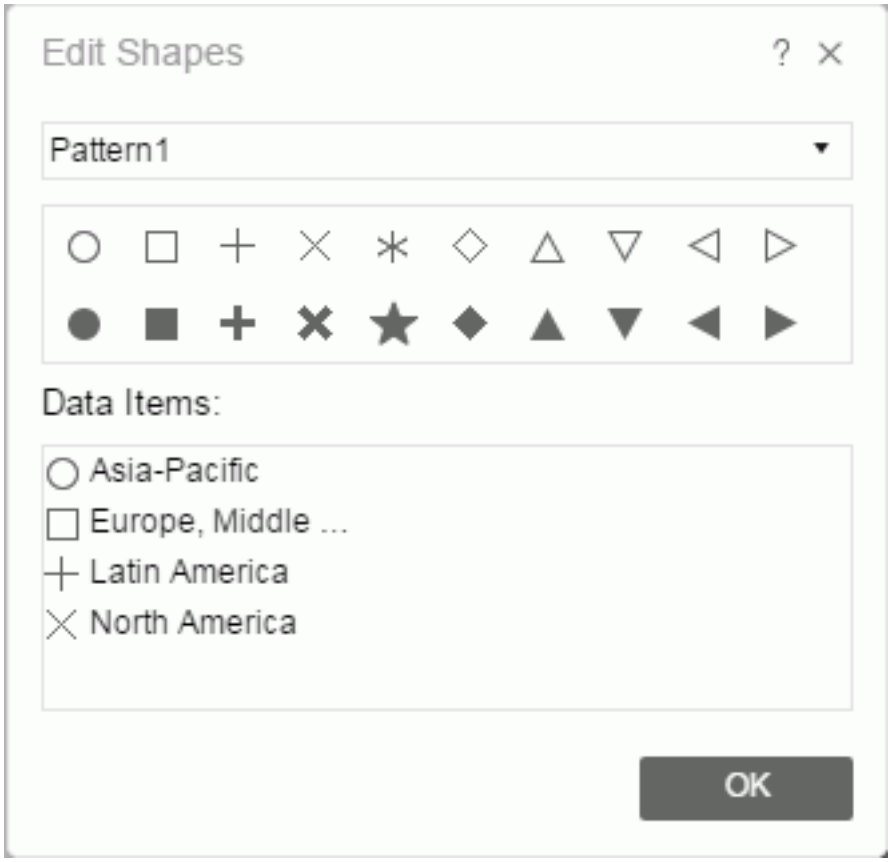
Displays the help document about this feature.

×

Ignores the setting and closes this dialog.

Edit Shapes dialog

This dialog helps you to specify shapes for the members of the shape legend field.



Pattern

After selecting a pattern from the drop-down list, a preview of the shapes of the pattern is provided below the pattern name. Changing a pattern will apply the shapes in the new pattern to all of the listed field members.

Data Items

Lists the members of the field with the shapes in the selected pattern.

To customize the shape of a data item, select the data item and then select a shape in the pattern preview box.

OK

Closes this dialog and applies the specified shapes.

?

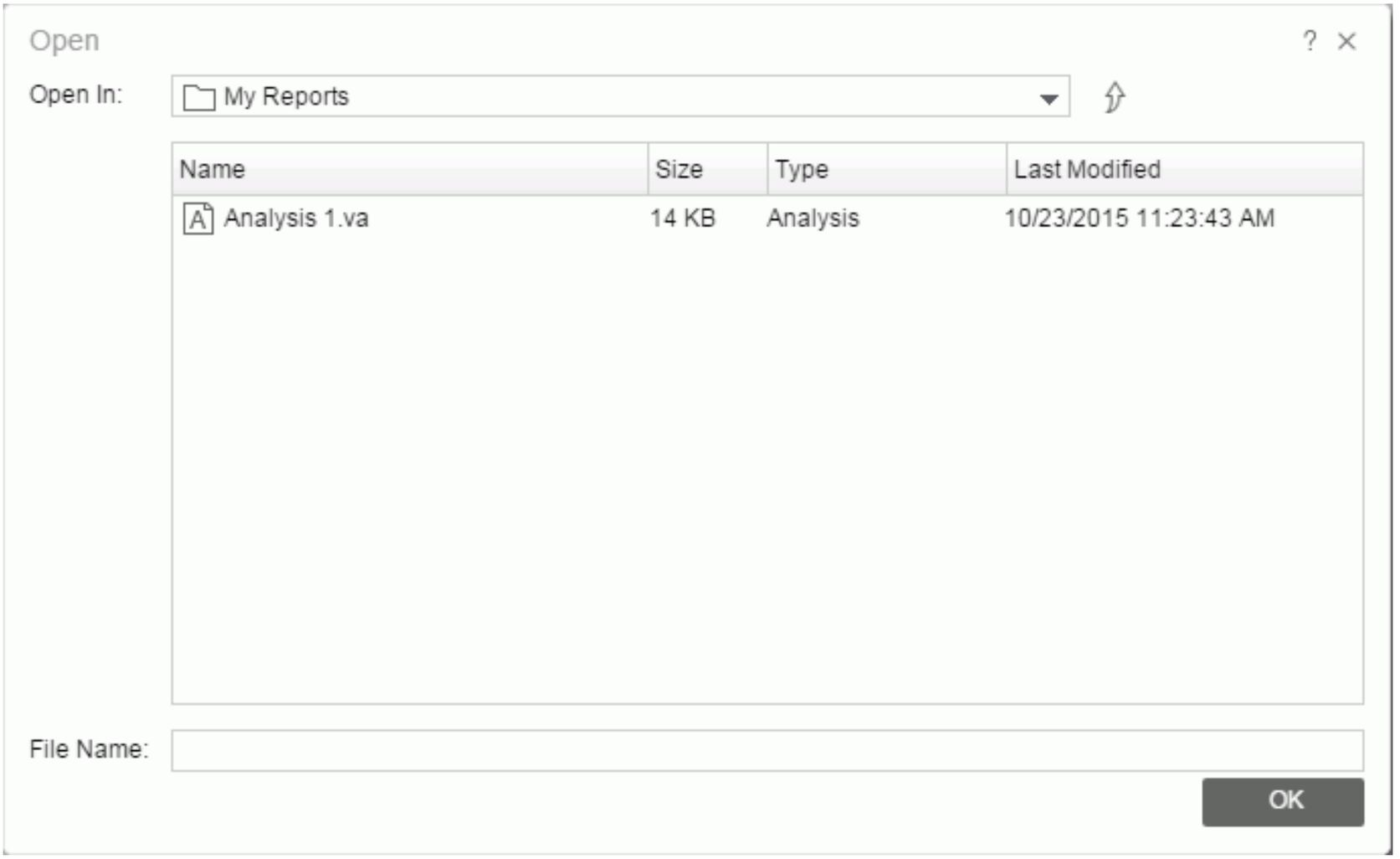
Displays the help document about this feature.

×

Ignores the setting and closes this dialog.

Open dialog

This dialog helps you to open another Visual Analysis template in a new Visual Analysis window.



Open In

Browse to the directory in the server resource tree where the target Visual Analysis template is located and then select it. Use the button to go to the parent folder.

The resource table shows the Visual Analysis templates in the current directory.

File Name

Shows the selected Visual Analysis template file name.

OK

Closes this dialog and opens the selected Visual Analysis template.

?

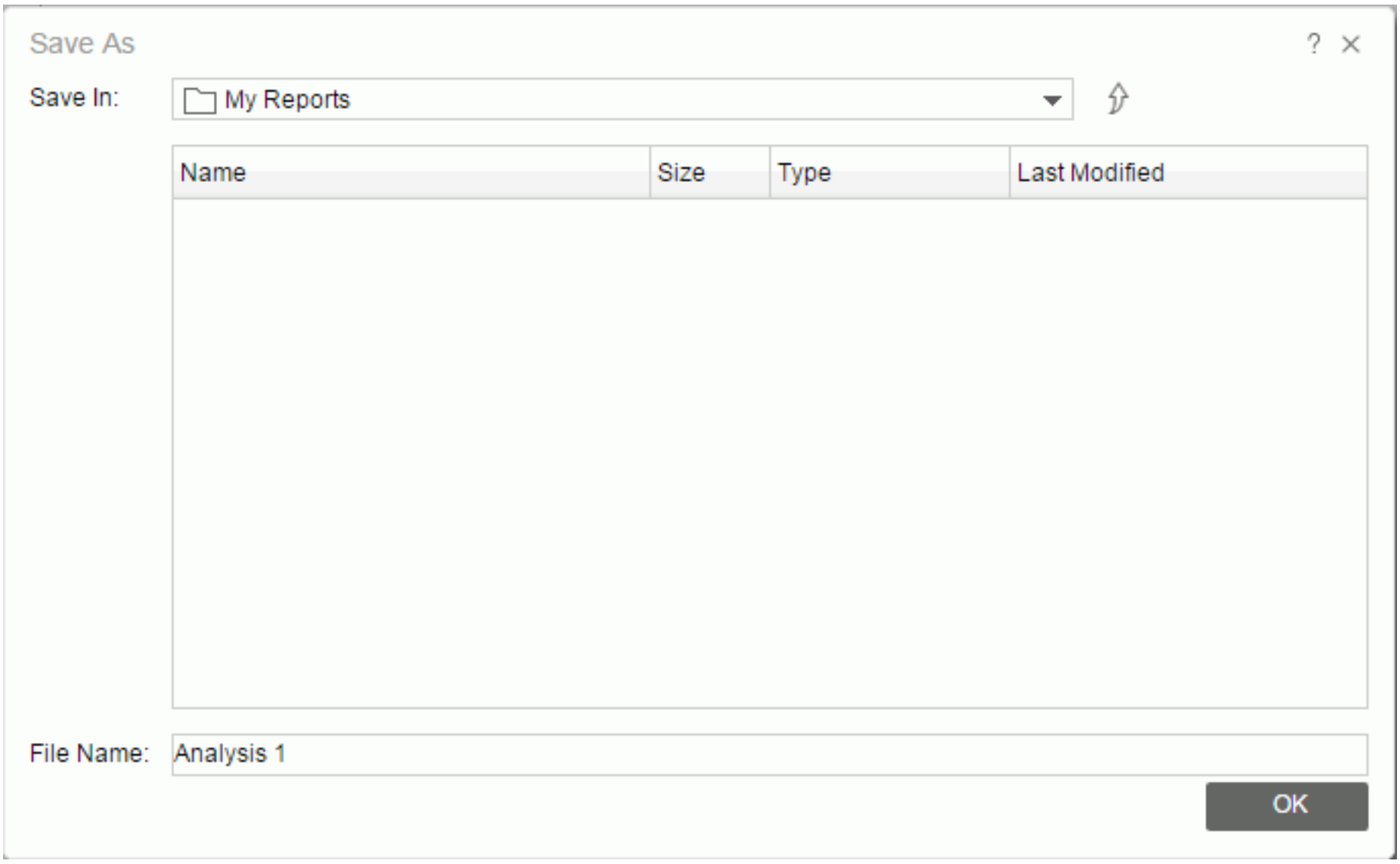
Displays the help document about this feature.

×

Ignores the setting and closes this dialog.

Save As dialog

This dialog is used to save the current Visual Analysis template with a different name or location.



Save In

Specifies the directory in the server resource tree where you want to save the Visual Analysis template. Use the button to go to the parent folder. The root folder cannot be used to store resources into.

Visual analysis results are saved in the My Reports and Public Reports folders together with reports.

The resource table shows the Visual Analysis templates in the current directory. Click the column names to change the order of the Visual Analysis templates in the table list if required.

- **Name**
  Displays the file names.
- **Size**
  Displays the file size.
- **Type**
  Displays the file type.
- **Last Modified**
  Displays the last modified time of the file.

File Name

Specifies the file name for the Visual Analysis template to save, without suffix.

OK

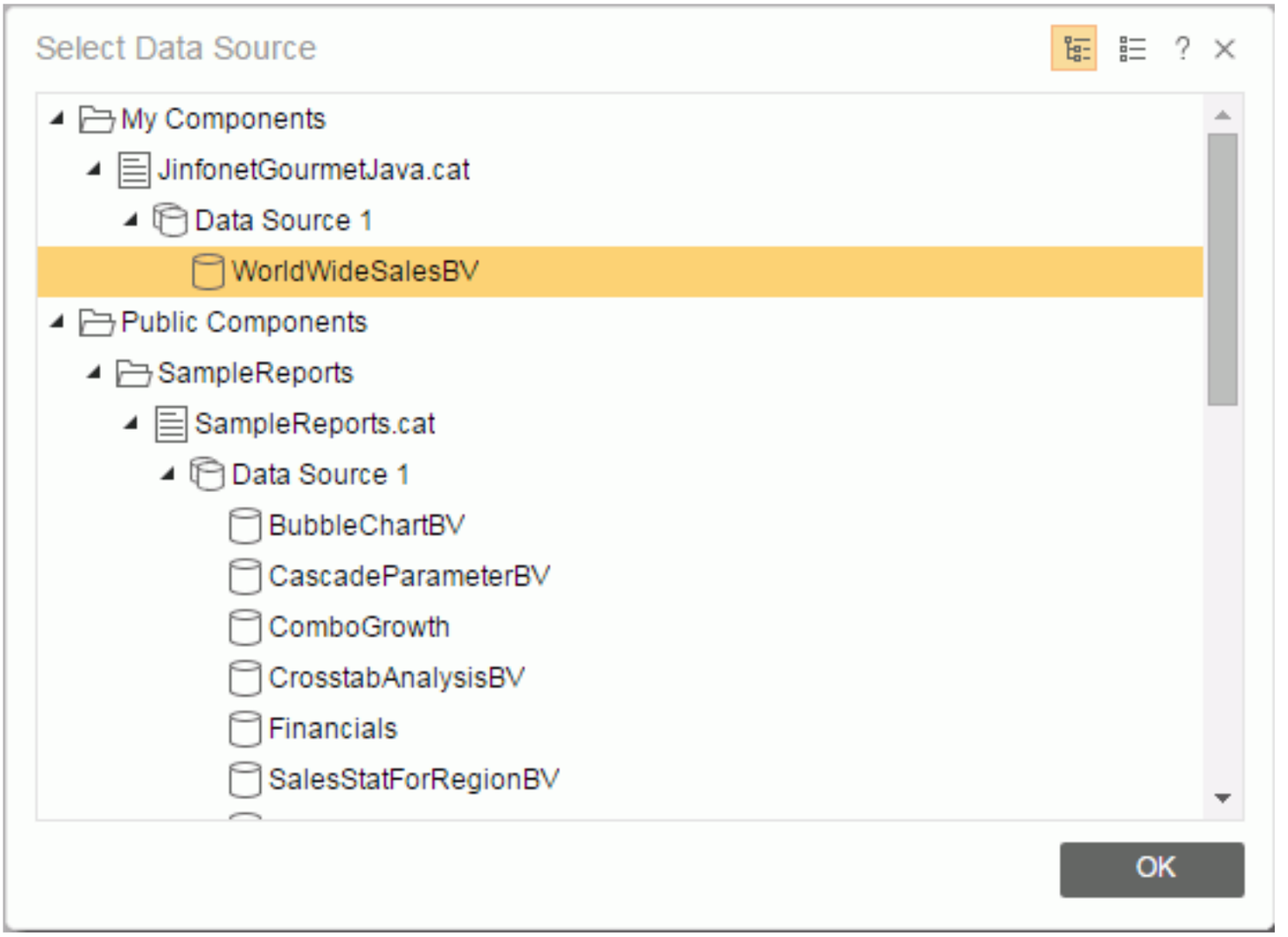
Closes this dialog and saves the Visual Analysis template.

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Select Data Source dialog

This dialog helps you to select a business view for performing visual analysis based on.



The available business views are listed for selection. The available business views come from the current catalog in the current directory in the server resource tree. If the current catalog contains no business view or if it is the server root directory, all available business views will be listed.

Displays the available business views in a tree structure.

Displays the available business views as list.

OK

Closes this dialog and opens Visual Analysis.

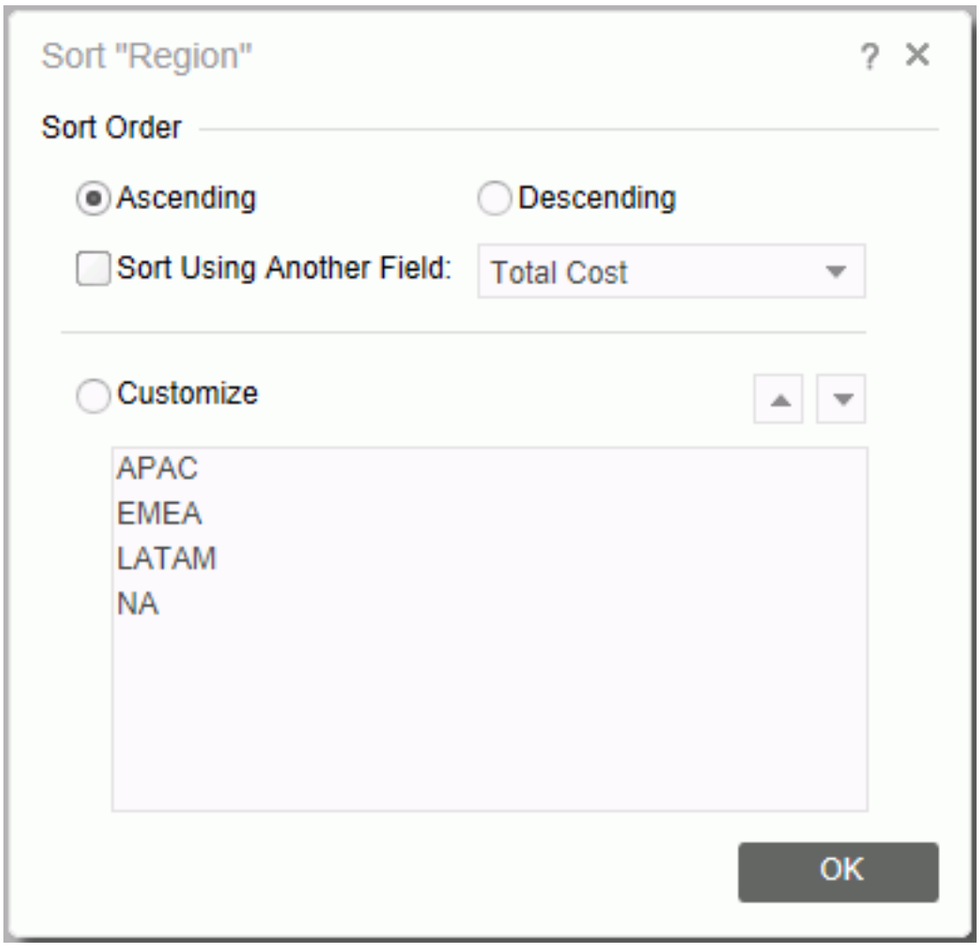
?

Displays the help document about this feature.

Ignores the setting and closes this dialog.

Sort dialog

This dialog helps you to sort the selected field.



Sort Order

It works when Customize is not selected.

- **Ascending**
  Sorts the field in the ascending order. If Sort Using Another Field is specified, it sorts the specified field in the ascending order.
- **Descending**
  Sorts the field in the descending order. If Sort Using Another Field is specified, it sorts the specified field in the descending order.
- **Sort Using Another Field**
  If checked, you can select an aggregation field from the drop-down list that follows to perfom the sort on.

Customize

Adjusts the order of the members of the current field manually. The field members are listed in the text box below. Select a member and then click the proper arrow button to move it up or down.

OK

Closes this dialog and applies the sorting order.

?

Displays the help document about this feature.

×

Ignores the setting and closes this dialog.

Formula Reference

A formula is a symbolic statement of the manipulations you want to perform on certain data. JReport provides a number of built-in functions for you to build your formula. Also it provides user defined formula (UDF) functions that allow you to design your own formula functions.

Generally, formulas can be used:

- to calculate information you cannot obtain directly from database data fields,
- to compare data,
- to join text with data,
- to convert data from one form to another,
- to enhance the formatting options with text strings, and
- to do a number of other things to customize your reports.

Specifying Java compiler for compiling formulas in the JRE

When running in the Java VM environment using JDK, JReport Server can use javac.exe in the JDK to compile formulas. However, if JReport runs with JRE, you need to specify a Java compiler for compiling formulas, otherwise, formulas in reports may not work or even result in the failure of running the reports. In the JRE environment, if you are going to use Eclipse to compile formulas, add the option *-Djavacompiler=ECJcompiler* in the startup file of JReport Server. If you are to use tools.jar to compile, copy the tools.jar from `JDK(where the JRE is located)\lib` to the `JRE\lib` folder and add the option *-Djavacompiler=SysCompiler* in the JReport Server startup file.

This chapter contains the following topics:

- Formula levels
- Formula syntax
- Data type
- Built-in functions
- Operators

Formula levels

Formulas can be categorized into 6 levels. These levels determine when the actual statements in your formula will be executed. JReport Engine will make several passes over the data to determine when formulas are executed. Even though an individual formula appears to be a certain level, if it references anything in the higher level then it also becomes the higher level.

- **Constant level formula**
  A constant level formula is a formula that does not refer to any DBFields, summaries, global variables, page level special fields (the value of the special field is ready at the time when the report result is generated) or other non-constant formulas. For example, `return CurrentDate()` is a constant formula. JReport will execute these formulas before fetching data from the dataset, so if you place it in the report header, it will be executed immediately as it is encountered.

- **Record level pass one formula**
  A formula is a record level pass one formula if and only if the following are true:
  - Refers to a DBField or any other formula which references a DBField.
  - Does not refer to any page level special fields or formulas which refer to page level special fields.
  - Does not refer to any global variable or any record level pass two formulas
  - Does not refer to a summary field.

  Record level pass one formulas are executed on each record immediately after the record is fetched from the dataset. They are often used in the detail panel of banded reports and tables such as calculating a value based on other DBFields such as `return @PRICE - @COST`.

- **Record level pass two formula**
  A formula is a record level pass two formula if and only if the following are true:
  - Refers to a DBField and a summary.
  - Refers to any other record level pass two formula.

  Record level pass two formulas are executed on each record of the dataset but only after the data has been grouped. For example, if a record level pass two formula is placed in the banded header, the group is considered to be the entire report. If it is placed in a group header such as Category, it will be the value after all of the detail rows have been processed for the group. For example, you might want to print in a group footer the group name and total such as `return "Total for " + @CATEGORY + ": " + @Sum_Total;`.

- **Group level formula**
  A formula is a group level formula if it refers to only summaries or other group level formulas. For example, if you have a summary called Sum_Total and a summary called Count_Detail, you could write a record level pass two formulas as `return @Sum_Total / @Count_Detail` to print the average total sales on each line of the order.

  These formulas are executed when breaking the group on which the summaries are defined which could be a group or the entire report for a report level summary in the banded header or banded footer.

- **Report global and global formulas**
  If a formula refers to a global variable or report global variable and does not refer to any page level special fields, there are several rules for when the formula will be calculated.
  - Report Header and Group Header: Formula will be calculated when the group starts.
  - Page Header, Page Footer and Detail: Formula will be calculated on each record after the data has been grouped the same as record level pass two formulas.
  - Report Footer and Group Footer: Formula will be calculated when ending the group or report.

  These formulas will be calculated based on all of the records of the group or report. For example, if you are calculating an accumulated total that you display in a report footer or group footer, then it will work as expected. For example, you may want to calculate a total margin such as:

  ```
  global number GrossProfit = 0;
  GrossProfit = GrossProfit + (@Price - @Cost);
  return GrossProfit;
  ```

  However, if you are showing a running total in the detail, you will see the group or report total on every line rather than the expected running total. The solution is to convert the formula to a page level formula by adding a reference to any page level special fields (see below).

- **Page level formula**
  All of the other formula types are converted to a page level formula if the formula refers to next(), prev() or any of the page level special fields. Page level special fields include FetchDate, FetchTime, RecordNumber, PageNumber, PrintDate, and PrintTime.

  The page level formulas are calculated in the exact sequence of the panels as they are laid out in JReport Engine. Thus if a page level formula includes any of the page level special fields, it will be calculated in the detail or group header and footer exactly in the order they are laid out. In the above example, to do a running total on each detail the code is as follows:

  ```
  PageNumber;
  global number GrossProfit = 0;
  PageNumber;
  GrossProfit = GrossProfit + (@Price - @Cost);
  return GrossProfit;
  ```

  PageNumber also controls the sequence of calculations around page breaks. If PageNumber is referenced, the calculation will be done before the page break. If PageNumber is not referenced, the calculation will be done after the page break. This is important when using formulas for setting properties.

Formula syntax

JReport formula syntax is a syntax with simple flow control and other statements. The Formula Editor is where you write your formula statements. You should follow JReport formula syntax when writing formulas.

In this syntax, sequence and selection control structure are supported. You can use it to control the execution of the statements. In this syntax, a parameter can be used as variable, and its value can be used in computation or making decisions. Since the value of the parameter is input before the report is run, the report can respond to user input and generate different results.

Statements

The statement is the smallest executable unit in the syntax. A SQL procedure is a sequence of statements. Statements are separated by semicolon (;). There are many kinds of statements, such as declare statement, expression statement, assignment statement, if-else statement, and return statement.

Declare statement

The declare statement is used to declare variables.

- **Syntax**: `VariableType VariableName;`

  Example: `String sql;`

  This statement declares a local String type variable called sql.

- **Syntax**: `global VariableType VariableName;`

  Example: `global integer i;`

  This statement declares a global Integer type variable called i.

Expression statement

The expression statement is used to calculate the value of an expression.

Assignment statement

The assignment statement is used to calculate the expression on the right side of assignment operator and assign the value to the variable on the left side.

Syntax: `Variable = Expression`

The type of value of expression must be compatible with the type of the variable.

Example: `total = amount * price`

In this example, where the total, amount and price are declared variable. The statement calculate the "amount * price" and assign the value to "total".

If - else statement

The if-else statement provides conditional control of an action. Normally, the statements are executed one by one sequentially. The if-else statement enables alternative actions that depend on the evaluation result of the expression.

Syntax:

```
if (expression) {
  statement(s);}
or
if (expression) {
  statement(s);}
else {
  statement(s);}
```

Where, the expression must be a logical expression and have a Boolean value. The if clause checks the condition presented by the logical expression. If the condition is true, the statement following the if will be executed. Otherwise, the statement following the else will be executed.

Example:

```
If (score >= 60) {
    return "PASS";}
else {
    return "FAIL";}
```

In this example, if the score is greater than or equal to 60, the result is pass, otherwise the result is fail.

Return statement

The return statement is used to stop the execution of procedure and return a value if necessary. The return statement is the last statement executed.

Syntax: `return` or `return expression;`

The expression will be calculated before the procedure returns.

Example: `return results;`

For statement

The for statement provides a compact way to iterate over a range of values.

Syntax:

```
for (initialization; termination; increment) {
  statement;
};
```

Or

```
for (initialization; termination; increment) statement;
```

The `initialization` is an expression that initializes the loop - it's executed once at the beginning of the loop. The `termination` expression determines when to terminate the loop. This expression is evaluated at the top of each iteration of the loop. When the expression evaluates to false, the loop terminates. Finally, `increment` is an expression that gets invoked after each iteration through the loop. All these components are optional.

Examples:

```
integer i=0, j=0;
For(i=0;i<100;i=i+1){
    j=j+1;
};
return j;
```

Or

```
integer i=0, j=0;
For(i=0;i<100;i=i+1) j=j+1;
return j;
```

Both examples will return 100 as the result.

```
integer siteint=0;
string stite="&_isMultiple_jrs.param$Site=true";

if(@P_int==0) then
    stite=stite+"&jrs.param$Site=%07"
else {
    for(siteint=0;siteint<@P_int;siteint=siteint+1)
    stite= stite+"&jrs.param$Site=" + @P_int
};
return stite;
```

While statement

The while statement is used to continually execute a block of statements while a condition remains true.

Syntax:

```
while (expression) do {
  statement;
};
```

Or

```
while(expression) do statement;
```

First, the while statement evaluates `expression`, which must return a Boolean value. If the expression returns true, then the while statement executes the `statement(s)` associated with it. The while statement continues testing the expression and executing its block until the expression returns false.

Example:

```
integer i=0;
While(i<100) do {
    i=i+1;
};
return i;

Or
integer i=0;
While(i<100) do i=i+1;
return i;
```

Both examples will return 100 as the result.

Note: JReport Designer provides another statement that is similar to the while statement - the do-while statement.

syntax:

```
do { statement(s); } while (expression);
```

Or

```
do statement while(expression);
```

Instead of evaluating the expression at the top of the loop, do-while evaluates the expression at the bottom. Thus the statements associated with a do-while are executed at least once.

Example:

```
integer j=0;
Do {
    j=j+1;}
while(j<100);
return j;

Or
integer j=0;
Do j=j+1 while(j<100);
return j;
```

Both examples will return 100 as the result.

Select statement

The select statement is usually used in the case that the value of a single variable may determine one of a number of different choices.

A select statement is given a variable and compares its value to all cases in the switch; if there is a case that matches the value, all statements in the matching case are executed. If none match, and a default is given, all statements following the default keyword are executed.

Syntax:

```
select (variable name){
  case expression_1: statement
  case expression_2: statement
  ...
  default: statement
};
```

The `expression_1` and `expression_2` should be variables or constants. The statement in each should be a single statement or multi-statement (compound statement). When multi-statement is used, they must be enclosed by {}.

Notes:

- The value of the select is only evaluated one time even when a case modifies the tested value.
- No break statement is required, the logic flow does not continue through the following case statements.

Examples:

```
integer i = 0 ; string j = 'a';
i = @"Customers_Customer ID";
select(i) {
    case 1: j='bb'
    case 2: j='cc'
    default: j ='dd'
};
return j;



string state="",result="";
if (IsNull(@Customer_State)) then
    state="Others"
else
    state = @Customer_State;

select (state) {
    case  "BC","ID","MT","WA":result="Northwest"
    case  "CA","AZ","TX","NM":result="Southwest"
    case  "ME","MA","NH","NY":result="Northeast"
    case  "FL","NC","GA","SC":result="Southeast"
    default: result="Rest of Country"
};
return result;
```

Expression

The expression is a combination of values, operators and functions that produce a result. The value in the expression can be a literal value or a variable, while the operator defines the operation between values; the function is used to perform an action and return a value.

Value

The value specified in an expression can be a literal value or a variable value. JReport formula syntax supports seven data types of value. See the section Data type for details.

Literal value

The Literal Value is a value that is used exactly as it is displayed. A literal value represents a value such as a Number, String, or Date. For example, "Name", 98.6...

Variables

A variable is a named storage unit used to store a value. The identifier (name) of variable can be used to refer to the value of the variable or refer to the storage space of the variable. In an expression, the identifier is used to refer to the value, and in assignment statement, the identifier on the left is used to refer to the storage space.

- **Undeclared variables**
  The undeclared variables include parameter, DBField, special field and summary defined in the catalog. They can be used as a variable in a formula.
- **Declared variables**
  Another type of variable which must be declared in a declare statement before use. The declared variable is a real variable and can be assigned in the assignment statement. There are two kinds of declared variables:
  - **Local Variable**: Variable which is only valid in the formula where it is declared.
  - **Global Variable**: Variable which is valid in all the formulas once it is declared.

Report level global variable

To control the formula more flexibly, JReport offers you one more global variable scope to operate besides component level global variable. That is report level global variable. You can use "report" as the keyword to define the global variable in a formula.

For the two levels of global variable, the following definitions will help you distinguish them:

- Component level: Its lifetime is the same as the component's lifetime. In its lifetime, whenever a formula is executed, the variable will be executed if it is used in the formula.
- Report level: Its lifetime is the same as the report's (primary report) lifetime. Even in different components (using same data source), the same report level global variable will have only one instance. The report level variable can be passed to different components (using same data source). The components include banded objects, tables, charts, tabulars, crosstabs, and subreports (including many layers of subreports).

When using the keyword "global" to define a global variable in a formula, if you add no words before the keyword, the global variable is on the component level; if you add "report" before the keyword, the global variable will be a report level global variable.

Example

This example uses the report global variable to do a calculation in the report. First, create three formulas based on the report global variable as follows:

Formula1: report global number a =0;

Formula2: a = a+1;

Formula3: return a;

Then use them in one report including two BandedObjects(BandedObject1 and BandedObject2) in the same dataset as follows:

1. Insert Formula1, Formula2, Formula3 to the BandedObject1(15 records) as follows:

   BandedHeader:Formula1

   DetailPanel:Formula2

   BandedFooter:Formula3 - returns 15.

2. Then insert Formula2 and Formula3 to the BandedObject2(25 records) as follows:

   DetailPanel:Formula2

   BandedFooter:Formula3 - returns 40.

Notes:

- For nested data containers, report level global formulas may lead to an unexpected result that the calculations are done at the end of the group or report. If you add PageNumber or other page level special field in the formula's definition, the generated result will occur in the specified order. For more information, see Formula levels.
- Although one report can have two or more data sources, report level global variable cannot span across data sources, otherwise the formula will not run correctly, thus, even the report global variables with the same variable name in different data sources are different in one report.
- If there is a report level global variable report global integer g, and it is used both in the primary report and subreport, then this variable g can communicate with each other.
- If you want to calculate the record number of the subreport and primary report separately, you can define a different report level variable from the primary report, but if one subreport is inserted into one primary report more than once you will be unable to correctly use the variable separately, in such circumstance, you can change the report level variable into component level variable to solve the problem.

Operator

An operator is a symbol that indicates an operation to be performed on values. The JReport procedure language has several classes of operators, including math, comparison and logical operators. For detailed information about operator, see the section Operators in this chapter.

Function

Functions return a value as its result. JReport formula syntax provides many built-in functions. You can refer to the section Built-in functions for detailed information about function.

Data type

Type				Description
Numeric	**Integer**	**int**	32-bit two's complement	Defines a field containing a whole number between -2,147,483,648 and 2,147,483,647. Fractional values inserted into an Integer field are truncated. If specified as NOT NULL WITH DEFAULT, null values will be replaced by zero.
		long	64-bit two's complement	A data type that holds large integers.
		BigInt		Immutable arbitrary-precision integers.
	Real Number	**float**	32-bit IEEE 754	This data type defines an 32bit-IEEE standard 754 floating-point field. Floating-point fields can be defined as single or double (default) precision based on the value of Integer. If the value of integer is between 1 and 21, the type is single precision floating-point, which requires 4 bytes of storage. If the value of integer is between 22 and 53 inclusive, the type is double precision floating-point, which requires 8 bytes of storage. If a field is specified as NOT NULL WITH DEFAULT, null values will be replaced by zero.
		double	64-bit IEEE 754	This data type defines an 64-bit IEEE standard 754 floating-point field. Floating-point fields can be defined as single or double (default) precision based on the value of integer. If the value of integer is between 22 and 53 inclusive, the type is double precision floating-point, which requires 8 bytes of storage. If a field is specified as NOT NULL WITH DEFAULT, null values will be replaced by zero.
		currency		This data type contains a dollar amount between; -10^{12} and 10^{12} and is displayed in a user-defined format.

Type			Description
DateTime	**Date**	MM/dd/yy MM/dd/yyyy	Dates designate a point in time according to the Gregorian calendar. Historical dates may not follow this calendar. The standard input format for this data type is MM/dd/yy or MM/dd/yyyy (automatic conversion is performed between these two formats).
	Time	HH:mm:ss hh:mm:ss	The Time field data type is defined by the standard input format of HH:mm:ss (using hour in day 0--23) or hh:mm:ss (using in am/pm 1--12).
	TimeStamp	yyyy-MM-dd hh:mm:ss	This combination Date and Time data type is displayed in the format yyyy-MM-dd-HH: mm.ss.nnnnnnn.
Other Types	**Char**	16-bit Unicode character	A single character.
	String		A data type that holds character information.
	Boolean	true or false	True or false.

Notes:

Format	Description
yyyy	An Integer from 0001 to 9999, representing a year.
MM	An Integer from 1 to 12 representing a month
dd	An Integer from 1 to 31 (maximum depends on the month and year) representing the day of the month.
HH	An Integer from 0 to 23, representing the hour in day.
hh	An Integer from 1 to 12, representing the hour in am/pm.
mm	An Integer from 0 to 59, representing minutes.
ss	An Integer from 0 to 59 representing seconds (default 0).
nnnnnn	An Integer (up to 6 digits) representing microseconds. If any trailing digit is omitted, 0 is assumed.

Built-in functions

The Functions panel in the Formula Editor lists all the built-in functions JReport provides. They are powerful and commonly used.

The following are types of the built-in functions:

- Array
- Date Range
- Date/Time
- Financial
- Math
- String
- Others

Array

The Array functions are all used to summarize field data in several different ways.

- Any()
- Average()
- Count()
- DistinctCount()
- Every()
- Maximum()
- Minimum()
- PopulationStdDev()
- PopulationVariance()
- StdDev()
- Sum()
- UBound()
- Variance()

Any()

Function

- Any(array_variable);
- Any(field_variable);

Description

This function is used to check whether there is at least one element in the argument is true. If there is, return true; if there isn't, return false.

Arguments

- **field_variable**
  field_variable is a special variable in formula. It refers to a group of values of a database field or of a formula field.
- **array_variable**
  array_variable refers to an array.

Return type

Returns a Boolean value.

Examples

- Boolean x = Any([false, false, false, true, false, false])
- Boolean x = Any(@isExcellent)
- Boolean x = Any(@formula1) (@formula is a formula which returns a Boolean array.)
- Boolean x = Any(@@ctf1) (@@ctf1 is a crosstab formula which returns a Boolean array.)

Average()

Function

- Average(array_variable);
- Average(field_variable);
- Average(field_variable, groupby);

Description

This function is used to return the average value of a group of values referred to by the first argument. The second argument, if there is one, returns the group by field name.

Arguments

- **field_variable**
  field_variable is a special variable in Formula. It refers to a group of values of a database field or of a formula field. Whatever source they come from, database or formula, they should be of Number or Currency data type.
- **array_variable**
  array_variable refers to an array whose element is of the Number or Currency value.
- **groupby**
  groupby can be a constant String which indicates the groupby field name or a field variable on which grouping occurs, or a parameter variable that can accept a different value before each execution of a report.

Return type

The result type relies on the first argument. It can be of the Number or Currency data type, if the data type of each element in the first argument is also of the Number or Currency data type respectively.

Examples

- Number x = Average([1.0, 2.0, 3.0, 4.5, 45.0, 67.0])
- Number x[6] = [1.0, 2.0, 3.0, 4.5, 45.0, 67.0]; Number y = Average(x)
- Currency x = Average([$1, $2, $3.0, $4.5, $45, $67])
- Currency x[] = [$1, $2, $3.0, $4.5, $45, $67]; Currency y = Average(x)
- Number x = Average(@dbfield)
- Number x = Average(@formula)
- Number x = Average(@dbfield, "group_field")
- Number x = Average(@dbfield, @"group_field")
- Number x = Average(@dbfield, @parameter)
- Currency x = Average(@dbfield)
- Currency x = Average(@formula)
- Currency x = Average(@formula, "group_field")
- Currency x = Average(@formula, @"group_field")
- Currency x = Average(@formula, @parameter)

Note: If a field_variable is assigned to a variable, for example X, the variable X will lose the characteristic of representing a group of values. The following formula will then be treated as incorrect.

```
Number x = @dbfield;
Number y = Average(x); // system will prompt you that there is no such kind of function.
```

Count()

Function

- Count(array_variable);
- Count(field_variable);
- Count(field_variable, groupby);

Description

This function is used to count the number of a group of values referred to by the first argument. The second argument, if there is one, returns the group by field name.

Arguments

- **field_variable**
  field_variable is a special variable in Formula. It refers to a group of values of a database field or of a formula field.
- **array_variable**
  array_variable refers to an array.
- **groupby**
  groupby can be a constant String to indicate the groupby field name, or a field variable on which the grouping occurs, or a parameter variable that can accept a different value before each execution of a report.

Return type

Returns a Number value.

Examples

- Number x = Count([1.0, 2.0, 3.0, 4.5, 45.0, 67.0]).
- Number x[6] = [1.0, 2.0, 3.0, 4.5, 45.0, 67.0];
  Number y = Count(x).
- Number x = Count(["str1","str2","str3","str4","str5","str6"]).
- Number x = Count(@dbfield).
- Number x = Count(@formula).
- Number x = Count(@dbfield, "group_field").
- Number x = Count(@formula, @"group_field").
- Number x = Count(@dbfield, @parameter).

Note: If a field_variable is assigned to a variable, for example X, the variable X will lose the characteristic of representing a group of values. The following formula will then be treated as in correct.

```
Number x = @dbfield;
Number y = Count(x); // system will prompt you that there is no such kind of function.
```

DistinctCount()

Function

- DistinctCount(array_variable);
- DistinctCount(field_variable);
- DistinctCount(field_variable, groupby);

Description

This function is used to count the number of distinct values referred to by the first argument together. The second argument, if there is one, gives the group by field name.

Arguments

- **field_variable**
  field_variable is a special variable in Formula, and refers to a group of values from a database or formula field.
- **array_variable**
  array_variable refers to an array whose element should be a Number.
- **groupby**
  groupby can be a constant String to indicate the groupby field name, or a field variable on which grouping occurs, or a parameter variable that can accept a different value before each execution of a report.

Return type

Returns a Number value.

Examples

- Number x = DistinctCount([1.0, 1.0, 3.0, 4.5, 4.5, 67.0]) // result is 4
- Number x[6] = [1.0, 1.0, 3.0, 4.5, 4.5, 67.0];
  Number y = DistinctCount(x)
- Number x = DistinctCount(["str1","str1","str3","str4","str4","str6"])}
- Number x = DistinctCount(@dbfield)
- Number x = DistinctCount(@formula)
- Number x = DistinctCount(@dbfield, "group_field")
- Number x = DistinctCount(@formula, @"group_field")
- Number x = DistinctCount(@dbfield, @parameter)

Note: If a field_variable is assigned to a variable, for example X, the variable X will lose the characteristic of representing a group of values. The following formula will then be treated as incorrect.

```
Number x = @dbfield;
Number y = DistinctCount(x); // system will prompt you that there is no such kind of
function.
```

Every()

Function

- Every(array_variable);
- Every(field_variable);

Description

This function is used to check whether each element in the argument is true. If yes, return true; if not, return false.

Arguments

- **field_variable**
  field_variable is a special variable in formula. It refers to a group of values of a database field or of a formula field.
- **array_variable**
  array_variable refers to an array.

Return type

Returns a Boolean value.

Examples

- Boolean x = Every([false, false, false, true, false, false])
- Boolean x = Every(@isExcellent)
- Boolean x = Every(@formula1) (@formula is a formula which returns a Boolean array.)
- Boolean x = Every(@@ctf1) (@@ctf1 is a crosstab formula which returns a Boolean array.)

Maximum()

Function

- Maximum(array_variable);
- Maximum(field_variable);
- Maximum(field_variable, groupby);

Description

This function is used to pick up the maximum value from a group of values referred to by the first argument. The groupby argument specifies the group by field name.

Arguments

- **field_variable**
  field_variable is a special variable in Formula, and refers to a group of values from a database or formula field.
- **array_variable**
  array_variable refers to an array whose element should be Number.
- **groupby**
  Groupby can be a constant String to indicate the groupby field name, or a field variable on which grouping occurs, or a parameter variable that can accept a different value before each execution of a report.

Return type

The result type relies on the first argument. It may be Number, Currency, String, Date, Time, DateTime, or Boolean if the datatype of each element in the first argument is Number, Currency, String, Date, Time, DateTime or Boolean respectively.

Examples

- Number x = Maximum([1.0, 2.0, 3.0, 4.5, 45.0, 67.0])
- Number x[6] = [1.0, 2.0, 3.0, 4.5, 45.0, 67.0];
  Number y = Maximum(x)
- Number x = Maximum(@dbfield)
- Date x = Maximum(@formula)
- Time x = Maximum(@dbfield, "group_field")
- DateTime x = Maximum(@dbfield, @"group_field")
- String x = Maximum(@dbfield, @parameter)
- Currency x = Maximum(@dbfield)
- Boolean x = Maximum(@formula)
- Date x = Maximum(@formula, "group_field")
- Currency x = Maximum(@formula, @"group_field")
- Currency x = Maximum(@formula, @parameter)
- String x = Maximum(['1', '2', '3.0', '4.5', '45', '67'])
- Date x = Maximum([toDate(1998, 1, 1), toDate(1997, 7, 1), toDate(1999, 1, 16)])

Note: If a field_variable is assigned to another variable, for example X, the variable X will lose the characteristic of representing a group of values. The following formula will then be treated as incorrect.

```
Number x = @dbfield;
Number y = Maximum(x); // system will prompt you that there is no such kind of function.
```

Minimum()

Function

- Minimum(array_variable);
- Minimum(field_variable);
- Minimum(field_variable, groupby);

Description

This function is used to pick up the minimum value from a group of values referred to by the first argument together. The second argument, if there is one, gives the group by field name.

Arguments

- **field_variable**
  field_variable is a special variable in Formula, and refers to a group of values from a database or formula field.
- **array_variable**
  array_variable refers to an array whose element should be a Number.
- **groupby**
  groupby can be a constant String to indicate the groupby field name, or a field variable on which grouping occurs, or a parameter variable that can accept a different value before each execution of a report.

Return type

The result type relies on the first argument. It may be Number, Currency, String, Date, Time, DateTime, or Boolean if the data type of each element in the first argument is Number, Currency, String, Date, Time, DateTime or Boolean respectively.

Examples

- Number x = Minimum([1.0, 2.0, 3.0, 4.5, 45.0, 67.0])
- Number x[6] = [1.0, 2.0, 3.0, 4.5, 45.0, 67.0];
  Number y = Minimum(x)
- Number x = Minimum(@dbfield)
- Date x = Minimum(@formula)
- Time x = Minimum(@dbfield, "group_field")
- DateTime x = Minimum(@dbfield, @"group_field")
- String x = Minimum(@dbfield, @parameter)
- Currency x = Minimum(@dbfield)
- Boolean x = Minimum(@formula)
- Date x = Minimum(@formula, "group_field")
- Currency x = Minimum(@formula, @"group_field")
- Currency x = Minimum(@formula, @parameter)
- String x = Minimum(['1', '2', '3.0', '4.5', '45', '67'])
- Date x = Minimum([toDate(1998, 1, 1), toDate(1997, 7, 1), toDate(1999, 1, 16)]

Note: If a field_variable is assigned to a variable, for example X, the variable X will lose the characteristic of representing a group of values. The following formula will then be treated as incorrect.

```
Number x = @dbfield;
Number y = Minimum(x); // system will prompt you that there is no such kind of function.
```

PopulationStdDev()

Function

- PopulationStdDev(array_variable);
- PopulationStdDev(field_variable);
- PopulationStdDev(field_variable, groupby);

Description

This function is used to find the population standard deviation of a group of values referred to by the first argument. The second argument, if there is one, gives the group by field name.

Arguments

- **field_variable**
  field_variable is a special variable in Formula, and refers to a group of values from a database or formula field. Whatever kind of source the data comes from, database or formula, it should be of the Number data type.
- **array_variable**
  array_variable refers to an array whose element should be Number value.
- **groupby**
  groupby can be a constant String to indicate the groupby field name, or a field variable on which grouping occurs, or a parameter variable that can accept a different value before each execution of a report.

Return type

Returns a Number value.

Examples

- Number x = PopulationStdDev([1.0, 2.0, 3.0, 4.5, 45.0, 67.0])
- Number x[6] = [1.0, 2.0, 3.0, 4.5, 45.0, 67.0];
  Number y = PopulationStdDev(x)
- Number x = PopulationStdDev(@dbfield)
- Number x = PopulationStdDev(@formula)
- Number x = PopulationStdDev(@dbfield, "group_field")
- Number x = PopulationStdDev(@dbfield, @"group_field")
- Number x = PopulationStdDev(@dbfield, @parameter)

Note: If a field_variable is assigned to a variable, for example X, the variable X will lose the characteristic of representing a group of values. The following formula will then be treated as incorrect.

```
Number x = @dbfield;
Number y = PopulationStdDev(x); // system will prompt you that there is no such kind of
function.
```

PopulationVariance()

Function

- PopulationVariance(array_variable);
- PopulationVariance(field_variable);
- PopulationVariance(field_variable, groupby);

Description

This function is used to find the population variance of a group of values referred to be the first argument. The second argument, if there is one, gives the group by field name.

Arguments

- **field_variable**
  field_variable is a special variable in Formula, and refers to a group of values from a database or formula field. Whatever kind of source the data comes from, database or formula, it should be of the Number data type.
- **array_variable**
  array_variable refers to an array whose element should be Number value.
- **groupby**
  groupby can be a constant String to indicate the groupby field name, or a field variable on which grouping occurs, or a parameter variable that can accept a different value before each execution of a report.

Return type

Returns a Number value.

Examples

- Number x = PopulationVariance([1.0, 2.0, 3.0, 4.5, 45.0, 67.0])
- Number x[6] = [1.0, 2.0, 3.0, 4.5, 45.0, 67.0];
  Number y = PopulationVariance(x)
- Number x = PopulationVariance(@dbfield)
- Number x = PopulationVariance(@formula)
- Number x = PopulationVariance(@dbfield, "group_field")
- Number x = PopulationVariance(@dbfield, @"group_field")
- Number x = PopulationVariance(@dbfield, @parameter)

Note: If a field_variable is assigned to a variable, for example X, the variable X will lose the characteristic of representing a group of values. The following formula will then be treated as incorrect.

```
Number x = @dbfield;
Number y = PopulationVariance(x); // system will prompt you that there is no such kind
of function.
```

StdDev()

Function

- StdDev(array_variable);
- StdDev(field_variable);
- StdDev(field_variable, groupby);

Description

This function is used to find the standard deviation of a group of values referred to by the first argument. The second argument, if there is one, gives the group by field name.

Arguments

- **field_variable**
  field_variable is a special variable in Formula, and refers to a group of values from a database or formula field. Whatever kind of source the data comes from, database or formula, it should be of the Number data type.
- **array_variable**
  array_variable refers to an array whose element should be a Number value.
- **groupby**
  groupby can be a constant String to indicate the groupby field name, or a field variable on which grouping occurs, or a parameter variable that can accept a different value before each execution of a report.

Return type

Returns a Number value.

Examples

- Number x = StdDev([1.0, 2.0, 3.0, 4.5, 45.0, 67.0])
- Number x[6] = [1.0, 2.0, 3.0, 4.5, 45.0, 67.0];
  Number y = StdDev(x)
- Number x = StdDev(@dbfield)
- Number x = StdDev(@formula)
- Number x = StdDev(@dbfield, "group_field")
- Number x = StdDev(@dbfield, @"group_field")
- Number x = StdDev(@dbfield, @parameter)

Note: If a field_variable is assigned to a variable, for example X, the variable X will lose the characteristic of representing a group of values. The following formula will then be treated as incorrect.

```
Number x = @dbfield;
Number y = StdDev(x); // system will prompt you that there is no such kind of function.
```

Sum()

Function

- Sum(array_variable);
- Sum(field_variable);
- Sum(field_variable, groupby);

Description

This function is used to add a group of values (referred to by the first argument) together. The second argument, if there is one, gives the group by field name.

Arguments

- **field_variable**
  field_variable is a special variable in Formula, and refers to a group of values from a database or formula field. Whatever kind of source the data comes from, database or formula, it should be of the Number or Currency data type.
- **array_variable**
  array_variable refers to an array whose element should be Number or Currency value.
- **groupby**
  groupby may be a constant String to indicate the groupby field name, or a field variable on which grouping occurs, or a parameter variable that can accept a different value before each execution of a report.

Return type

The result type relies on the first argument. It may be Number or Currency if the data type of each element in the first argument is Number or Currency respectively.

Examples

- Number x = Sum([1.0, 2.0, 3.0, 4.5, 45.0, 67.0])
- Number x[6] = [1.0, 2.0, 3.0, 4.5, 45.0, 67.0];
  Number y = Sum(x)
- Currency x = Sum([$1, $2, $3.0, $4.5, $45, $67])
- Currency x[] = [$1, $2, $3.0, $4.5, $45, $67];
  Currency y = Sum(x)
- Number x = Sum(@dbfield)
- Number x = Sum(@formula)
- Number x = Sum(@dbfield, "group_field")
- Number x = Sum(@dbfield, @"group_field")
- Number x = Sum(@dbfield, @parameter)
- Currency x = Sum(@dbfield)
- Currency x = Sum(@formula)
- Currency x = Sum(@formula, "group_field")
- Currency x = Sum(@formula, @"group_field")
- Currency x = Sum(@formula, @parameter)

Note: If a field_variable is assigned to a variable, for example X, the variable X will lose the characteristic of representing a group of values. The following formula will then be treated as incorrect.

```
Number x = @dbfield;
Number y = Sum(x); // system will prompt you that there is no such kind of function.
```

UBound()

Function

- UBound(array_variable);

Description

This function is used to return the number of elements in an array.

Argument

- **array_variable**
  array_variable refers to an Array value, expression or variable.

Return value

Returns a Number value.

Examples

- The return value of the following statement is 5.

  ```
  integer a[] = [1,1,1,1,1];
  UBound(a)
  ```

- The return value of the following statement is 6.

  UBound([$1, $2, $3.0, $4.5, $45, $67]).

- ```
  if(isNull(@"Customer Name"))then
  {
  return "No Data";
  }
  else{
  string s[]=StringSplit(@"Customer Name");
  string s1="Java";
  string s2[]=filter(s,s1);
  if (UBound(s2)!=0) then
  return right(s2[0],4)
  else
  return "No matching data ";
  }
  ```

  This formula searchs for the customer names containing the word "Java".

Variance()

Function

- Variance(array_variable);
- Variance(field_variable);
- Variance(field_variable, groupby);

Description

This function is used to find the variance of a group of values referred to by the first argument. The second argument, if there is one, gives the group by field name.

Arguments

- **field_variable**
  field_variable is a special variable in Formula, and refers to a group of values from a database or formula field. Whatever kind of source the data comes from, database or formula, it should be of the Number data type.
- **array_variable**
  array_variable refers to an array whose element should be Number value.
- **groupby**
  groupby can be a constant String to indicate the groupby field name, or a field variable on which grouping occurs, or a parameter variable that can accept a different value before each execution of a report.

Return type

Returns a Number value.

Examples

- Number x = Variance([1.0, 2.0, 3.0, 4.5, 45.0, 67.0])
- Number x[6] = [1.0, 2.0, 3.0, 4.5, 45.0, 67.0];
  Number y = Variance(x)
- Number x = Variance(@dbfield)
- Number x = Variance(@formula)
- Number x = Variance(@dbfield, "group_field")
- Number x = Variance(@dbfield, @"group_field")
- Number x = Variance(@dbfield, @parameter)

Note: If a field_variable is assigned to a variable, for example X, the variable X will lose the characteristic of representing a group of values. The following formula will then be treated as incorrect.

```
Number x = @dbfield;
Number y = Variance(x); // system will prompt you that there is no such kind of
function.
```
```

Date Range

Note that all the following functions cannot be used to return a value singly.

Aged0To30Days()
Aged31To60Days()
Aged61To90Days()
AllDatesFromToday()
AllDatesFromTomorrow()
AllDatesToToday()
AllDatesToYesterday()
Calendar1stHalf()
Calendar1stQtr()
Calendar2ndHalf()
Calendar2ndQtr()
Calendar3rdQtr()
Calendar4thQtr()
Last4WeeksToSun()
Last7Days()
LastFullMonth()
LastFullWeek()
LastYearMTD()
LastYearYTD()
MonthToDate()
Next30Days()
Next31To60Days()
Next61To90Days()
Next91To365Days()
Over90Days()
WeekToDateFromSun()
YearToDate()

Aged0To30Days()

This function returns a range of Data values that is within the previous 30 days from today. For example, if today is 10/14/99, the return range is from 09/15/99 to 10/14/99.

Example

The return value of the following statement is Not Expired.

```
 if (ToDate(1999,9,20) in Aged0To30Days())
   return "Not Expired"
 else
   return "Expired"
```

Aged31To60Days()

This function returns a range of Data values that is within the previous 31 to 60 days from today date. For example, if today is 10/13/99, the return range is from 08/12/99 to 09/13/99.

Example

If today is 10/13/99, the return value of the following statement is false.

```
return ToDate(1998, 4, 20) in Aged31To60Days()
```

Aged61To90Days()

This function returns a range of Data values that is within the previous 61 to 90 days from today date. For example, if today is 10/13/99, the return range is from 07/13/99 to 08/11/99.

Example

If today is 10/13/99, the return value of the following statement is false.

```
return ToDate(1998, 4, 20) in Aged61To90Days()
```

AllDatesFromToday()

This function returns a range of Data values that include any date from the present day to a date about 10000 years in the future Data value. For example, if today is 04/15/98, the return range is from 04/15/98 to the future.

Example

If today is 10/10/99, the return value of the following statement is false.

```
return ToDate(1998, 4, 20) in AllDatesFromToday()
```

AllDatesFromTomorrow()

This function returns a range of Data values that include any date from tomorrow to any future Data value. For example, if today is 04/15/98, the return range is from 04/16/98 to the future.

Example

If today is 10/10/99, the return of the following statement is false.

```
return ToDate(1998, 4, 20) in AllDatesFromTomorrow()
```

AllDatesToToday()

This function returns a range of Data values that include all dates up through the present day. For example, if today is 4/15/98, the return range is from the past to 04/15/1998.

Example

If today is 10/10/99, the return value of the following statement is true.

```
return ToDate(1998, 4, 20) in AllDatesToToday()
```

AllDatesToYesterday()

This function returns a range of Data values that include all dates up through the previous day. For example, if today is 4/15/98, the return range is from the past to 04/14/1998.

Example

If today is 10/10/99, the return value of the following statement is true.

```
return ToDate(1998, 4, 20) in AllDatesToYesterday()
```

Calendar1stHalf()

This function returns a range of Data values that include all dates from January 1st through June 30th.

Example

If today is 10/10/99, the return value of the following statement is false.

```
return Today() in Calendar1stHalf()
```

Calendar1stQtr()

This function returns a range of Data values that include all dates from January 1st through to March 31st.

Example

If today is 10/10/99, the return value of the following statement is false.

```
return Today() in Calendar1stQtr()
```

Calendar2ndHalf()

This function returns a range of Data values that include all dates from July 1st through to December 31st.

Example

If today is 10/10/99, the return value of the following statement is true.

```
return Today() in Calendar2ndHalf()
```

Calendar2ndQtr()

This function returns a range of Data values that include all dates from April 1st through to June 30th.

Example

If today is 10/10/99, the return value of the following statement is false.

```
return Today() in Calendar2ndQtr()
```

Calendar3rdQtr()

This function returns a range of Data values that include all dates from July 1st through to September 30th.

Example

If today is 10/10/99, the return value of the following statement is false.

```
return Today() in Calendar3rdQtr()
```

Calendar4thQtr()

This function returns a range of Data values that include all dates from October 1st through December 31st.

Example

If today is 10/10/99, the return value of the following statement is true.

```
return Today() in Calendar4thQtr()
```

Last4WeeksToSun()

This function returns a range of Data values that include the four weeks previous to last Sunday. For example, if today is 4/15/95, the return range is from 3/13/95 to 4/9/95.

Example

If today is 10/10/99, the return value of the following statement is false.

```
return ToDate(1999, 4, 20) in Last4WeeksToSun()
```

Last7Days()

This function returns a range of Data values that include all dates from seven days ago to today (including today). For example, if today is 04/15/98, the return range is from (include) 04/09/98 to 04/15/98.

Example

If today is 10/10/99, the return value of the following statement is false.

```
return ToDate(1999, 4, 20) in Last7Days()
```

LastFullMonth()

This function returns a range of Data values that include all dates from the first to the last day of the previous month. For example, if today is 04/15/98, the return range is from 03/01/98 to 03/31/98.

Example

If today is 10/10/99, the return value of the following statement is false.

```
return ToDate(1999, 4, 20) in LastFullMonth()
```

LastFullWeek()

This function returns a range of Data values that include all dates from the Sunday to the Saturday of the previous week. For example, if today is 04/10/95, the return range is from 04/02/95 to 04/08/95.

Example

If today is 10/10/99, the return value of the following statement is false.

```
return ToDate(1999, 4, 20) in LastFullWeek()
```

LastYearMTD()

This function returns a range of Data values that fall on any date in the current month last year, up to the current date last year. For example, if today is 04/15/98, the return range is from 04/01/97 to 04/15/97.

Example

If today is 10/10/99, the return value of the following statement is false.

```
return ToDate(1999, 4, 20) in LastYearMTD()
```

LastYearYTD()

This function returns a range of Data values that include all dates in the last year, up to the current date last year. For example, if today is 04/15/98, the return range is from 01/01/97 to 04/15/97

Example

If today is 10/10/99, the return value of the following statement is true.

```
return ToDate(1998, 4, 20) in LastYearYTD()
```

MonthToDate()

This function returns a range of Data values that include all dates from the first day of the month to today. For example, if today is 04/15/98, the return range is from 04/01/98 to 04/15/98.

Example

If today is 10/10/99, the return value of the following statement is true.

```
return ToDate(1999, 10, 2) in MonthToDate()
```

Next30Days()

This function returns a range of Data values that include all days in the next 30 days starting from today (including today). For example, if today is 04/15/98, the return range is from 04/15/98 to 05/15/98.

Example

If today is 10/10/99, the return value of the following statement is false.

```
return ToDate(1999, 10, 2) in Next30Days()
```

Next31To60Days()

This function returns a range of Data values that include all days in the next 31 to 60 days starting from today. For example, if today is 04/15/98, the return range is from 05/16/98 to 06/14/98.

Example

If today is 10/10/99, the return value of the following statement is false.

```
return ToDate(1999, 10, 2) in Next31To60Days()
```

Next61To90Days()

This function returns a range of Data values that include all days in the next 61 to 90 days starting from today. For example, if today is 04/15/98, the return range is from 06/15/98 to 07/14/98.

Example

If today is 10/10/99, the return value of the following statement is false.

```
return ToDate(1999, 4, 20) in Next61To90Days()
```

Next91To365Days()

This function Returns a range of Data values that include all days in the next 91 to 365 days starting from today. For example, if today is 04/15/98, the return range is from 07/15/98 to 04/15/99.

Example

If today is 10/10/99, the return value of the following statement is false.

```
return ToDate(1999, 4, 20) in Next91To365Days()
```

Over90Days()

This function returns a range of Data values that include all days that are more than 90 days older than the current date. For example, if today is 04/15/98, the return range is from the past to 01/14/98.

Example

If today is 10/10/99, the return value of the following statement is true.

```
return ToDate(1999, 4, 20) in Over90Days()
```

WeekToDateFromSun()

This function returns a range of Data values that include all days from last Sunday to today (including today). For example, if today is 04/13/95, the return range is from 04/09/95 to 04/13/95.

Example

If today is 10/10/99, the return value of the following statement is false.

```
return ToDate(1999, 4, 20) in WeekToDateFromSun()
```

YearToDate()

This function returns a range of Data values that include all days from the first day of the calendar year to today (including today). For example, if today is 04/15/98, the return range is from 01/01/98 to 04/15/98.

Example

If today is 10/10/99, the return value of the following statement is true.

```
return ToDate(1999, 4, 20) in YearToDate()
```

Date/Time

Date functions allow you to convert numbers to dates and dates to numbers.

- CurrentDate()
- CurrentDateTime()
- CurrentTime()
- DateAdd()
- DateDiff()
- DatePart()
- DateTimeTo2000()
- DateTimeToDate()
- DateTimeToSeconds()
- DateTimeToTime()
- DateTo2000()
- Day()
- DayOfWeek()
- DayOfYear()
- DTSTo2000()
- DTSToSeconds()
- FirstDayOfMonth()
- FirstDayOfQuarter()
- FirstDayOfWeek()
- FirstDayOfYear()
- FirstSundayOfMonth()
- FirstSundayOfYear()
- ForEachDay()
- ForEachHalfMonth()
- ForEachHalfYear()
- ForEachMonth()
- ForEachQuarter()
- ForEachWeek()
- ForEachYear()
- Hour()
- IsDate()
- IsDateTime()
- IsTime()
- Minute()
- Month()
- MonthName()
- Now()
- Quarter()
- Second()
- SelectedDate()
- Timer()
- ToDate()
- ToDateTime()
- Today()
- ToTime()
- WeekdayName()
- WeekFrom1970()
- WeekOfMonth()
- WeekOfYear()
- Year()

CurrentDate()

This function returns the current date on a report.

Return value

The return value is a date.

Example

If today is Oct. 6,1999, the return value of the following statement is 10/06/99.

```
CurrentDate()
```

CurrentDateTime()

This function returns the current date and time on a report.

Return value

The return value is a DateTime.

Example

If today is Oct.6,1999, the current time is 5:10:27 p.m., the return value of the following statement is 1999-10-06 05:10:27 PM.

```
CurrentDateTime()
```

CurrentTime()

This function returns the current time on a report.

Return value

The return value is a Time.

Example

If the current time is 5:10:27 p.m., the return value of the following statement is 05:10:27 PM.

```
CurrentTime()
```

DateAdd(intervalType, nIntervals, startDateTime)

Returns a DateTime value to which a specified number of time intervals have been added.

This function is used to add interval of time to a DateTime. Its main feature is that the DateTime returned will always be valid. For example, the function takes into account such factors as the number of days in a month and leap years. If you want to add or subtract days to a DateTime, you could use the addition and subtraction operators instead of the function with the "d" parameter. However, this function also handles other types of intervals such as adding months or hours to a DateTime.

Parameters

- intervalType - A String expression specifying the interval of time to be added.

IntervalType value	Description
yyyy	Year
q	Quarter (3-month period)
m	Month
y	Day of year
d	Day
w	Weekday
ww	Week (7-day period)
h	Hour
n	Minute
s	Second

- nIntervals - A Number or numeric expression specifying the number of intervals to be added. It can be positive (to get DateTimes from the future), or negative (to get DateTimes from the past).
- startDateTime - The DateTime value to which the intervals are to be added.

Return value

A DateTime value.

Examples

- `DateAdd("yyyy",1,CurrentDateTime())` - Returns the DateTime value for current datetime plus one year.
- `DateAdd("q",1,CurrentDateTime())` - Returns the DateTime value for current datetime plus three months.
- `DateAdd("m",1,CurrentDateTime())` - Returns the DateTime value for current datetime plus one month.
- `DateAdd("y",1,CurrentDateTime())` - Returns the DateTime value for current datetime plus one day.
- `DateAdd("d",1,CurrentDateTime())` - Returns the DateTime value for current datetime plus one day.
- `DateAdd("w",1,CurrentDateTime())` - Returns the DateTime value for current datetime plus one day.
- `DateAdd("ww",1,CurrentDateTime())` - Returns the DateTime value for current datetime plus seven days.
- `DateAdd("h",1,CurrentDateTime())` - Returns the DateTime value for current datetime plus one hour.
- `DateAdd("n",1,CurrentDateTime())` - Returns the DateTime value for current datetime plus one minute.
- `DateAdd("s",1,CurrentDateTime())` - Returns the DateTime value for current datetime plus one second.
- `DateAdd("d",-32,ToDateTime(1999,9,28))` - Returns the DateTime value 1999-8-27 00:00:00.
- `DateAdd("m",1,ToDateTime(1996,1,31))` - Returns the DateTime value 1996-2-29 00:00:00. Notice that DateAdd will not return the invalid value 1996-2-31 00:00:00.
- `DateAdd("q",17,ToDateTime(1999,9,28))` - Returns the DateTime value 2003-12-28 00:00:00.
- `DateAdd("h",-400,ToDateTime(1999,9,28))` - Returns the DateTime value 1999-9-11 08:00:00. In other words, this is the result of subtracting 400 hours from 1999-9-28 00:00:00.

Notes:

- To add days to a DateTime, you can use any of the interval type parameters "y", "d" or "w". They all have the same effect for DateAdd.
- DateAdd returns a DateTime value and not a Date value. However, you may need to convert this DateTime value to a Date value in certain situations (such as if you wanted to assign the value returned by DateAdd to a Date type variable). To convert to a Date value, use DateAdd in combination with the ToDate type conversion function. For example, the following returns the Date value for October 6, 1997.

  ```
  ToDate(DateAdd("yyyy",-2,ToDateTime(1999,10,6)))
  ```

DateDiff(intervalType, startDateTime, endDateTime, firstDayOfWeek)

Returns a number of time intervals between two specified dates.

Overload

- DateDiff(intervalType, startDateTime, endDateTime)
- DateDiff(intervalType, startDateTime, endDateTime, firstDayOfWeek)

Parameters

- intervalType - A string expression that is the interval of time you use to calculate the difference between startDateTime and endDateTime. Possible values can be:

IntervalType value	Description
Yyyy	Year
Q	Quarter
M	Month
Y	Day of year
D	Day (both "y" and "d" find the difference in days)
W	Number of weeks between startDateTime and endDateTime
Ww	Number of firstDayOfWeek's between startDateTime and endDateTime
H	Hour
N	Minute
S	Second

- startDateTime - The first DateTime value used in calculating the difference.
- endDateTime - The second DateTime value used in calculating the difference.
- firstDayOfWeek - An optional constant specifying the first day of the week. If not specified, jrSunday will be used.

Constant	Description
jrUseSystem	0 (Use the current system date)
jrSunday	1
jrMonday	2
jrTuesday	3
jrWednesday	4
jrThursday	5
jrFriday	6
jrSaturday	7

Return value

A Number value.

Examples

- Use DateDiff with the "d" or "y" interval type parameter to find the number of days between two dates.

  `DateDiff("d", ToDateTime(1999,10,7), ToDateTime(1999,10,10))` - Returns 3
- Use DateDiff with the "yyyy" interval type parameter to find the number of years difference between two dates. This use of DateDiff is the same as finding the difference between the year of endDateTime and the year of startDateTime.
  - `DateDiff("yyyy", ToDateTime(1999,10,7), ToDateTime(2005,2,10))` - Returns 6.
  - `DateDiff("yyyy", ToDateTime(1999,12,31), ToDateTime(2000,1,1))` - Returns 1 (a 1 year difference), even though there is only a 1 day difference between the dates.
  - `DateDiff("yyyy", ToDateTime(1999,1,1), ToDateTime(1999,12,31))` - Returns 0 (a 0 year difference), even though there is a 364 days difference.

  Suppose that for the above examples, the first date is the date that you bought a mutual fund, and the second date is the date you sold it. The mutual fund company must send you an annual report for every year in which you owned units in the fund. You would then get 7, 2 and 1 annual reports respectively, in the above cases.
- Use DateDiff with the "q" parameter to find the number of quarters (3 month periods) difference between two dates.
  - `DateDiff("q", ToDateTime(1999,10,6), ToDateTime(2003,5,20))` - Returns 14.
  - `DateDiff("q", ToDateTime(1999,3,31), ToDateTime(1999,4,1))` - Returns 1. The two dates are in adjacent quarters.
  - `DateDiff("q", ToDateTime(1999,1,1), ToDateTime(1999,3,31))` - Returns 0. The two dates are in the same quarter.

  Suppose the mutual fund company in the "yyyy" example mailed out quarterly reports. It would need to mail out 15, 2 and 1 quarterly reports respectively, in the above cases.
- Use DateDiff with the "m" parameter to find the number of months difference between two dates.

  `DateDiff("m", ToDateTime(1999,3,15), ToDateTime(1999,7,13))` - Returns 4.
- Use DateDiff with the "w" parameter to calculate the number of weeks between two dates. For example, if startDateTime is on a Tuesday, DateDiff counts the number of Tuesdays between startDateTime and endDateTime not including the initial Tuesday of startDateTime. Note however that it counts endDateTime if endDateTime is on a Tuesday.
  - `DateDiff("w", ToDateTime(1999,10,19), ToDateTime(1999,10,25))` - Returns 0.
  - `DateDiff("w", ToDateTime(1999,10,19), ToDateTime(1999, 10,26))` - Returns 1.

DatePart(intervalType, inputDateTime, firstDayOfWeek, firstWeekOfYear)

Returns a number that specifies a given part of a given date.

Overloads

- DatePart(intervalType, inputDateTime)
- DatePart(intervalType, inputDateTime, firstDayOfWeek)
- DatePart(intervalType, inputDateTime, firstDayOfWeek, firstWeekOfYear)
- DatePart(intervalType, inputTime)
- DatePart(intervalType, inputTime, firstDayOfWeek)
- DatePart(intervalType, inputTime, firstDayOfWeek, firstWeekOfYear)

Parameters

- intervalType - A String expression that specifies the part of a date to be returned. Possible values can be:

IntervalType value	Description
Yyyy	Extracts the year
Q	Quarter (the result is 1, 2, 3 or 4)
M	Month (the result is from 1 to 12)
Y	Day of year (1 to 365 or 366 in a leap year)
D	Day part of the date (1 to 31) Year
W	Day of week (1 to 7 with the result depending on firstDayOfWeek)
Ww	Week of year (1 to 53 with firstDayOfWeek and firstWeekOfYear determining the exact days of the first calendar week of the year)
H	Extracts the hour part of the given DateTime (0 to 23)
N	Minute part (0 to 59)
S	Second part (0 to 59)

- inputDateTime - The DateTime value whose part will be extracted.
- FirstDayOfWeek - An optional constant used to specify the first day of the week. If not specified, jrSunday will be used.

Constant	Description
jrUseSystem	0 (Use the current system date)
jrSunday	1
jrMonday	2
jrTuesday	3
jrWednesday	4
jrThursday	5
jrFriday	6
jrSaturday	7

- firstWeekOfYear - An optional constant specifying the first week of the year. If not specified, the first week is assumed to be the one in which Jan. 1 occurs (jrFirstJan1).

Constant	Value	Description
jrUseSystem	0	Start with week in which system date occurs
jrFirstJan1	1	Start with week in which January 1 occurs (default)
jrFirstFourDays	2	Start with the first week that has at least four days in the new year
jrFirstFullWeek	3	Start with first full week of the year

Examples

- `DatePart("d", ToDateTime(1999,8,15))` - Returns 15.
- `DatePart("m", ToDateTime(1999,8,15))` - Returns 8.
- `DatePart("n", ToDateTime(1999,8,15,10,35,0))` - Returns 35.
- `DatePart("q", ToDateTime(1999, 9,29))` - Returns 3 since September 29 is in the third quarter of the year.
- `DatePart("ww", ToDateTime(1997, 9,14))` - Returns 38 since 1997-9-14 is in the 38th week of 1997.

Notes:

- The DatePart function with "yyyy" intervalType parameter is the same as the Year function. Similarly, the DatePart function with "m", "d", "w", "h", "n" and "s" intervalType argument is the same as the functions Month, Day, Weekday (or DayOfWeek), Hour, Minute and Second respectively. On the other hand, there is no easy alternative to using the DatePart function for the "q", "y" and "ww" intervalType parameters.
- The firstDayOfWeek parameter affects the DatePart function when the interval type parameter is "w" or "ww". For all other intervalType parameter values, it is ignored.
- The firstWeekOfYear parameter affects the DatePart function only when the intervalType argument is "ww". For all other intervalType argument values, it is ignored.

DateTimeTo2000(Datetime, Number)

- **2 digit years(xx)**
  If the Year value is greater than the windowing number, 19 is appended before the 2 digits (19xx). If the Year value is less than or equal to the windowing number, 20 is appended (20xx).
- **4 digit years(19xx)**
  If the last two digits in the Year value are greater than the windowing number, the Year is retained as found in the Date field (19xx). If the two digits in the Year value are less than or equal to the windowing number, the first two digits are changed to 20 (20xx). If the first two digits in the year field are 20, the Year is retained as found in the Date field (20xx).

  > **Note:** If the year is less than or equal to 1899, and greater than or equal to 100, there will be no change to the date.

Parameters

- DateTime - Accepts only valid DateTime fields, with either 2 digit or 4 digit years.
- Number - A number between 0 and 99 corresponding with the desired windowing year.

Return value

A Date field with a four digit year.

Examples

- Here the window value is greater than the year, and so the year is changed to 20XX.

  `DateTimeTo2000(ToDateTime(1998, 12, 12, 3, 2, 1), 99)` - Returns 2098-12-12 3:02:01.
- Here the window value is less than the year, and so it will not change the year.

  `DateTimeTo2000(ToDateTime(1995, 1, 2, 3, 2, 1), 94)` - Returns 1995-01-02 3:02:01.
- These are examples of dates that will not be affected by this function.
  - `DateTimeTo2000(ToDateTime(1899, 12, 12, 5, 6, 7), 99)` - Returns 1899-12-12 5:06:07.
  - `DateTimeTo2000(ToDateTime(999, 12, 12, 5, 6, 7), 99)` - Returns 999-12-12 5:06:07.
- Some databases maintain the year of a Date as a two-digit field. These samples simulate that Date field.
  - `DateTimeTo2000(ToDateTime(93, 12, 12, 5, 6, 7), 96)` - Returns 2093-12-12 5:06:07.
  - `DateTimeTo2000(ToDateTime(98, 12, 12, 5, 6, 7), 50)` - Returns 1998-12-12 5:06:07.

DateTimeToDate(Datetime a)

This function evaluates the specified TimeStamp value and returns only the date.

Parameter

a - A TimeStamp value.

Return value

The return value is a date.

Example

If the current date is Oct. 17, 1999 and the current time is 12:27:15, the return value of the following statement is 10/17/99.

```
DateTimeToDate(CurrentDateTime())
```

DateTimeToSeconds(Datetime a)

This function evaluates the specified value and converts the time to the number of seconds from 00:00:00 to the specified time.

Parameter

a - A TimeStamp value.

Return value

The return value is an Integer.

Example

If the current date is Oct. 17, 1999 and the current time is 12:27:15, the return value of the following statement is 44835.

```
DateTimeToSeconds(CurrentDateTime())
```

DateTimeToTime(Datetime a)

This function evaluates the specified TimeStamp value and returns only the time.

Parameter

a - A TimeStamp value.

Return value

The return value is a Time.

Example

If the current date is Oct. 17, 1999 and the current time is 12:27:15, the return value of the following statement is 12:27:15.

```
DateTimeToTime(CurrentDateTime())
```

DateTo2000(date, number)

- 2 digit years(xx)
  If the Year value is greater than the windowing number, 19 is appended before the 2 digits (19xx). If the Year value is less than or equal to the windowing number, 20 is appended (20xx).
- 4 digit years(19xx)
  If the last two digits in the Year value are greater than the windowing number, the Year is preserved as found in the date field (19xx). If the last two digits in the Year value are less than or equal to the windowing number, the first two digits are changed to 20 (20xx). If the first two digits in the year field are 20, the Year is preserved as found in the date field (20xx).

Parameters

- date - Accepts only valid Date fields, with either 2 digit or 4 digit years.
- number - A number between 0 and 99 corresponding with the desired windowing year.

Return value

A Date field with a four digit year.

Examples

- `DateTo2000(ToDate(1993,12,12), 99)`- Returns 2093/12/12 because the window value is greater than the year, and so the date will be changed to 20XX.
- `DateTo2000(ToDate(1993,12,12), 92)`- Returns 1993/12/12 because the window value is less than the year, and so the year will not be changed.
- This is an example of a date that is not affected by this function:

  `DateTo2000(ToDate(1899,12,12), 99)`- Returns 1899/12/12.
- This is an example of a date that is not affected by this function.

  `DateTo2000(ToDate(100,12,12), 99)`- Returns 100/12/12.
- Some databases maintain the year of a date as a two-digit field. The following simulates this type of date field.
  - `DateTo2000(ToDate (98,12,12), 99)`- Returns 2098/12/12.
  - `DateTo2000(ToDate (98,12,12), 97)`- Returns 1998/12/12.
  - `DateTo2000(ToDate (9,12,12), 10)`- Returns 2009/12/12.
  - `DateTo2000(ToDate (1,12,12), 0)`- Returns 1901/12/12.

Notes:

- If the year is less than or equal to 1899 and greater or equal to 100, there will be no change to the date.
- Whether your database dates are interpreted as two digit years or 4 digit years (19xx) is dependent on the database driver you are using. The result of the formula will be the same in either case.

Day()

Day()

This function returns the day of the month.

Return value

The return value is an Integer.

Example

If today is Oct. 17, 1999, the return value of the following statement is 17.

```
Day()
```

Day(Date a)

This function extracts the day portion of a specified Date value.

Parameter

a - A Date value.

Return value

The return value is an Integer.

Example

If the current date is Oct. 17, 1999, the return value of the following statement is 17.

```
Day(CurrentDate())
```

Day(Datetime a)

This function extracts the day portion of a specified Date value.

Parameter

a - A TimeStamp value.

Return value

The return value is an Integer.

Example

If the current date is Oct. 17, 1999 and the current time is 12:27:15, the return value of the following statement is 17.

```
Day(CurrentDateTime())
```

DayOfWeek()

DayOfWeek()

This function returns the day of the week.

Return value

The return value is an Integer.

Example

If today is Thursday, the return value of the following statement is 5.

```
DayOfWeek()
```

DayOfWeek(Date a)

This function extracts the day portion of a specified Date value, and indicates which day of the week.

Parameter

a - A Date value.

Return value

The return value is an Integer.

Example

If today is Oct. 17, 1999 and today is Sunday, the return value of the following statement is 1.

```
DayOfWeek(CurrentDate())
```

DayOfWeek(Datetime a)

This function extracts the day portion of a specified Date value, and indicates which day of the week.

Parameter

a - A TimeStamp value.

Return value

The return value is an Integer.

Example

If the current date is Oct. 17, 1999, current time is 12:27:15 and today is Sunday, the return value of the following statement is 1 (Sunday is the first day of the week).

```
DayOfWeek(CurrentDateTime())
```

DayOfYear()

This function returns the day of the year.

Return value

The return value is an Integer.

Example

If today is Oct. 17, 1999, the return value of the following statement is 290.

```
DayOfYear()
```

DTSTo2000(DateString, Number)

- **2 digit years(xx)**
  If the Year value is greater than the windowing number, 19 is appended before the 2 digits(19xx). If the Year value is less than or equal to the windowing number, 20 is appended (20xx).
- **4 digit years(19xx)**
  If the last two digits in the Year value are greater than the windowing number, the Year is retained as found in the Date field (19xx). If the two digits in the Year value are less than or equal to the windowing number, the first two digits are changed to 20 (20xx). If the first two digits in the year field are 20, the Year is retained as found in the Date field (20xx).

  > **Note:** If the year is less than or equal to 1899 and greater than or equal to 100, there will be no change to the date.

Parameters

- Date String - Accepts only valid Date fields, with either 2 digit or 4 digit years. A DateTime string entered in the format: "yyyy/MM/dd HH: mm: ss.00" or "yy/MM/dd HH:mm:ss:00". For example, "1997/04/11 12:12:12.00" or "97/04/11 12:12:12.00".
- Number - A number between 0 and 99 corresponding with the desired windowing year.

Return value

A DateTime string with a four digit year.

Examples

- `DTSTo2000("1988-12-12 12:12:12", 90)`- Returns 2088-12-12 12:12:12 because the window value is greater than the year, and so the year will be changed to 20XX.
- `DTSTo2000("1988-12-12 12:12:12", 85)`- Returns 1988-12-12 12:12:12 because the window value is less than the year, and so the year will not change.
- These are examples of a date that will not be affected by the function:
  - `DTSTo2000("1899/12/12 1:2:3am", 99)`- Returns 1899/12/12 1:2:3am.
  - `DTSTo2000("100/12/12 1:02:03AM", 99)`- Returns 100/12/12 1:02:03AM.
- Some databases maintain the year of a date as a two-digit field and so these samples will simulate that date field:
  - `DTSTo2000("98/12/12 1:02:03AM", 99)`- Returns 2098/12/12 1:02:03AM.
  - `DTSTo2000("98/12/12 1:02:03", 97)`- Returns 1998/12/12 1:02:03AM.
  - `DTSTo2000("9/12/12 1:02:03AM", 10)`- Returns 2009/12/12 1:02:03AM.
  - `DTSTo2000("1/12/12 1:02:03AM", 0)`- Returns 1901/12/12 1:02:03AM.

DTSToSeconds(string)

Evaluates the specified string, and converts the Time value to the number of seconds from 00:00:00 (12:00 midnight) to the specified time.

Parameter

string - A String which includes a Date and a Time value.

Return value

Whole number.

Example

`DTSToSeconds("2004/06/11 13:42:15")`- Returns 49335.

FirstDayOfMonth(Date a)

This function returns the first day of the month.

Parameter

a - A Date value.

Return value

The return value is a Date.

Example

If the current date is 11/24/2008, the return value of the following statement is 11/01/2008.

```
FirstDayOfMonth(CurrentDate())
```

FirstDayOfQuarter(Date a)

This function returns the first day of the quarter.

Parameter

a - A Date value.

Return value

The return value is a Date.

Example

The return value of the following statement is 10/01/2008.

```
FirstDayOfQuarter(ToDate("11/24/2008"))
```

FirstDayOfWeek(Date a)

This function returns the first day of the week.

Parameter

a - A Date value.

Return value

The return value is a Date.

Example

The return value of the following statement is 11/23/2008.

```
FirstDayOfWeek(ToDate("11/24/2008"))
```

FirstDayOfYear(Date a)

This function returns the first day of the year.

Parameter

a - A Date value.

Return value

The return value is a Date.

Example

The return value of the following statement is 01/01/2008.

```
FirstDayOfYear(ToDate("11/24/2008"))
```

FirstSundayOfMonth(Date a)

This function returns the first Sunday of the month.

Parameter

a - A Date value.

Return value

The return value is a Date.

Example

The return value of the following statement is 11/02/2008.

```
FirstSundayOfMonth(ToDate("11/24/2008"))
```

FirstSundayOfYear(Date a)

This function returns the first Sunday of the year.

Parameter

a - A Date value.

Return value

The return value is a Date.

Example

The return value of the following statement is 01/06/2008.

```
FirstSundayOfYear(ToDate("11/24/2008"))
```

ForEachDay(Datetime a)

This function returns the Date portion of a specified DateTime value.

Parameter

a - A DateTime value.

Return value

The return value is a Date.

Example

If the DateTime is Dec.25,1999 20:12:50, the return value of the following statement is 12/25/99.

```
ForEachDay(ToDateTime(1999, 12, 25, 20, 12, 50))
```

ForEachHalfMonth(DateTime a)

If the day in the DateTime is less than 15, this function will return to the first day of the month. If the day is more than 15, it will return to the 16th day of the month.

Parameter

a - A DateTime value.

Return value

The return value is a date.

Examples

- If the datetime is Sep.13,1999 12:27:15, the return value of the following statement is 09/01/99.

  `ForEachHalfMonth(ToDateTime(1999,9,13,12,27,15))`
- If the datetime is Oct.18,1999 5:21:22, the return value of the following statement is 10/16/99.

  `ForEachHalfMonth(ToDateTime(1999,10,18,5,21,22))`

ForEachHalfYear(DateTime a)

If the month of a specified DateTime is before July, the function will return to Jan.1 of the year. Otherwise the function will return to July 1 of the year.

Parameter

a - A DateTime value.

Return value

The return value is a date.

Examples:

- If the datetime is May. 28,1999 10:10:25, the return value of the following statement is 01/01/99.

  `ForEachHalfYear(ToDateTime(1999, 5, 28, 10, 10, 25))`
- If the datetime is Nov. 15,1998 11:40:23, the return value of the following statement is 07/01/99.

  `ForEachHalfYear(ToDateTime(1998, 11, 15, 11, 40, 23))`

ForEachMonth(DateTime a)

This function returns the first day of the month in a specified DateTime value.

Parameter

a - A DateTime value.

Return value

The return value is a date.

Example

If the datetime is Sep.24,1999 12:05:56, the return value of the following statement is 09/01/99.

```
ForEachMonth(ToDateTime(1999,9,24,12,5,56))
```

ForEachQuarter(DateTime a)

If the month of a specified DateTime is before April, the function will return to Jan. 1 of the year. If the month is after April but before July, the function will return to April. 1 of the year. If the month is after July but before October, the function will return to July. 1 of the year. If the month is after October, the function will return to Oct.1 of the year.

Parameter

a - A DateTime value.

Return value

The return value is a date.

Examples

- If the datetime is Feb.14,1997 2:10:23, the return value of the following statement is 01/01/97.

  `ForEachQuarter(ToDateTime(1997,2,14,2,10,23))`
- If the datetime is May. 28,1999 10:10:25, the return value of the following statement is 04/01/99.

  `ForEachQuarter(ToDateTime(1999,5,28,10,10,25))`
- If the datetime is Aug. 28,1999 10:10:25, the return value of the following statement is 07/01/99.

  `ForEachQuarter(ToDateTime(1999,8,28,10,10,25))`
- If the datetime is Nov. 11,1998 11:40:23, the return value of the following statement is 10/01/98.

  `ForEachQuarter(ToDateTime(1998,11,11,11,40,23))`

ForEachWeek(DateTime a)

This function returns the Date of the first day of the week.

Parameter

a - A DateTime value.

Return value

The return value is a Date.

Example

If the DateTime is Oct. 15,1999 5:15:20 and it is Friday, the return value of the following statement is 10/10/99.

```
ForEachWeek(ToDateTime(1999, 10, 15, 5, 15, 20))
```

ForEachYear(DateTime a)

This function will return to Jan.1 of the year.

Parameter

a - A DateTime value.

Return value

The return value is a date.

Example

If the datetime is Aug. 28,1999 10:10:25, the return value of the following statement is 01/01/99.

```
ForEachYear(ToDateTime(1999, 8, 28, 10, 10, 25))
```

Hour()

Hour()

This function extracts the hour portion of the current time.

Return value

The return value is an integer.

Example

If the current time is 8:15:30, the return value of the following statement is 8.

```
Hour()
```

Hour(DateTime a)

This function extracts the hour portion of a specified TimeStamp value.

Parameter

a - A TimeStamp value.

Return value

The return value is an Integer.

Example

Suppose the datetime is Oct. 15,1999 9:35:22. The return value of the following statement is 9.

```
Hour(ToDateTime(1999,10,15,9,35,22))
```

Hour(Time a)

This function extracts the hour portion of a specified Time value.

Parameter

a - A Time value.

Return value

The return value is an Integer.

Example

Suppose the time is 10:05:26. The return value of the following statement is 10.

```
Hour(ToTime(10, 5, 26))
```

IsDate(number or string)

Returns true if the given String or Number value can be converted to a valid Date and returns false otherwise. A valid Date is any Date between 100-1-1 and 9999-12-31.

Parameters

- number - A Number value representing the number of days starting from January 1, 1900. It can be positive or negative, and is truncated if fractional.
- string - A text string representing a date and many forms can be accepted. If the string is a right Date value and format, the function returns true, otherwise false.

Return value

Boolean value, True or False.

Examples

- `IsDate(50)` - Returns true since the number 50 is interpreted as 50 days from Jan. 1, 1900, which is Feb 19, 1900.
- `IsDate(-50)` - Returns true since the number –50 is interpreted as 50 days before Jan 1, 1900, which is Nov 11, 1899.
- `IsDate("Feb 1, 2004")` - Returns true since the string is a right Date value and there is only one blank between 'Feb' and '1,' and '2004'.
- `IsDate("2004 - 2 - 19")` - Returns false since the String format is not right Date format.
- `IsDate("2003/2/29")` - Returns false since 2003 is not leap year.
- `IsDate("2004*2*19")` - Returns false since the String format is not right Date format.

IsDateTime(number/string)

Returns True if the given Number or String value can be converted to a valid DateTime, and returns False otherwise.

Parameters

- number - A Number value or expression to be tested for convertibility to a DateTime value. It can be positive, negative or fractional. It is interpreted as a number of days from Jan 1, 1900.
- string - A String value or expression to be tested for convertibility to a DateTime value. Many forms are accepted. If the given String value and format are right DateTime value and format, the function will return true, otherwise false.

Return value

Boolean value, true or false.

Examples

- `IsDateTime(15.2)` - Returns true since the given number argument is interpreted as the DateTime value 1900-01-15 04:48:00.

- `IsDateTime("Feb 23, 2004 15:23:25")` - Returns true.
- `IsDateTime("2004-2-23")` - Returns true.
- `IsDateTime("15:23:25")` - Returns true.

IsTime(number or string)

Returns True if the given Number or String value can be converted to a valid Time, and returns False if otherwise.

Parameters

- number - A Number value or expression to be tested for convertibility to a Time value. It can be positive, negative or fractional. It is interpreted as units of 24 hours. If the number makes the Time value between 0:0:0: and 23:59:59, the function will return true, otherwise false.
- string - A String value or expression to be tested for convertibility to a Time value. Many forms are accepted. If the String value and the format are right Time value and format, the function will return true, otherwise false.

Return value

Boolean value, true or false.

Examples

- `IsTime(0.5)` - Returns true since the given number argument is interpreted as 0.5 units of 24 hours, which is 12:00:00.
- `IsTime(3.8)` - Returns true since the given number is interpreted as 3.8 units of 24 hours, which is 19:12:00.
- `IsTime("8:30:00")` - Returns true since 8:30:00 is right Time value and format.
- `IsTime("8 30:00")` - Returns false since the given string is not right Time format.
- `IsTime("8:30:62")` - Returns false since the given string is not right Time value.

Minute()

Minute()

This function returns the minute portion of the current time.

Return value

The return value is an Integer.

Example

If the current time is 12:41:27, the return value of the following statement is 41.

```
Minute()
```

Minute(DateTime a)

This function returns the minute portion of a specified DateTime.

Parameter

a - A DateTime value.

Return value

The return value is an Integer.

Example

Suppose the datetime is Sep. 21,1997 12:20:25. The return value of the following statement is 20.

```
Minute(ToDateTime(1997, 9, 21, 12, 20, 25))
```

Minute(Time a)

This function returns the minute portion of a specified time.

Parameter

a - A Time value.

Return value

The return value is an Integer.

Example

Suppose the time is 8:30:27. The return value of the following statement is 30.

```
Minute(ToTime(8,30,27))
```

Month()

Month()

This function adding one returns the current month of the year.

Return value

The return value is an Integer, which ranges from 0 to 11.

Example

If the current month is September, the return value of the following statement is 8.

```
Month()
```

Month(Date a)

This function extracts the month portion from a specified Date value.

Parameter

a - A Date value.

Return value

The return value is an Integer.

Example

If the date is June 15,1999, the return value of the following statement is 6.

```
Month(ToDate(1999, 6, 15))
```

Month(DateTime a)

This function extracts the month portion from a specified TimeStamp value.

Parameter

a - A TimeStamp value.

Return value

The return value is an Integer.

Example

If the timestamp is July 15, 1999 10:10:10, the return value of the following statement is 7.

```
Month(ToDateTime(1999, 7, 15, 10, 10, 10))
```

MonthName(month, abbreviate)

Returns a String name for the specified month.

Parameters

- month - A whole number representing the month of the year, with values between 1 and 12, where 1 represents January.
- abbreviate - An optional Boolean value that indicates if the month name is to be abbreviated. If the abbreviate value is true, the function will return the abbreviated month name. Otherwise the function will return the full month name.

Return value

String value.

Examples

- `MonthName(4)` - Returns String value "April".
- `MonthName(12, true)` - Returns String value "Dec".
- `MonthName(12, false)` - Returns String value "December".

Now()

This function prints the current time on a report.

Return value

The return value is a Time.

Example

If the current time is 1:24:35 in the afternoon, the return value of the following statement is 1:24:35 PM.

```
Now()
```

Quarter()

Returns the quarter for the given date (1 for the first quarter).

Overloads

- Quarter()
- Quarter(Date,Integer)
- Quarter(Datetime,Integer)

Parameters

- Date - A Date value.
- Integer - An Integer value in the range of 1 to 12. It is an optional argument which indicates the beginning month of the first quarter. Specifying the number is useful for fiscal quarter. If b is not specified, the default is 1 which means January is the beginning month of the first quarter as same as the calendar quarter.
- Datetime - A DateStamp value.

Return value

Return an integer (1, 2, 3, 4).

Examples

- Quarter() - Return 3 if it is September according to the current system date.
- Quarter(ToDate(2006,11,28),4) - Return 3. In this example, the first quarter begins from the 4th month - April, so the 11th month is in the third quarter.
- Quarter(ToDateTime(2006,7,15,10,10,10),4) - Return 2.

Second()

Second()

This function extracts the second portion of the current time.

Return value

The return value is an Integer.

Example

If the current time is 4:55:03, the return value of the following statement is 3.

```
Second()
```

Second(Datetime a)

This function extracts the second portion of a specified TimeStamp.

Parameter

a - A TimeStamp value.

Return value

The return value is an Integer.

Example

Suppose the datetime is Oct. 15,1999 10:15:17, the return value of the following statement is 17.

```
Second(ToDateTime(1999, 10, 15, 10, 15, 17 ))
```

Second(Time a)

This function extracts the second portion of a specified time.

Parameter

a - A Time value.

Return value

The return value is an Integer.

Example

Suppose the time is 7:51:59, the return value of the following statement is 59.

```
Second(ToTime(7, 51, 59))
```

SelectedDate()

Returns the date selected in the calendar or returns today() when no calendar is available for selecting a date.

There is a built-in expression Selected Date in Calendar calling this function. The expression can be found in the Name drop-down list of the Calendar dialog.

Return value

Date value.

Example

If the calendar date is March 12, 2012, the function returns 03/12/2012.

Timer()

Returns the number of seconds elapsed since midnight.

Return value

Number value.

Example

If the current time is 9:28:30am, the function returns 34110.

ToDate()

ToDate(number or string)

Uses the function IsDate(number or string) to check whether the number/string is right or not. If the number/string is right, the function returns a Date value between 100-1-1 and 9999-12-31. If the number/string is wrong, it returns null.

Parameters

- number - A value representing the number of days starting from January 1, 1900. It can be positive or negative, and is truncated if fractional.
- string - A text string representing a date and many forms can be accepted.

Return value

Date value or null.

Examples

- `ToDate(50.5)` - Returns 1900-2-19 since fraction is truncated.
- `ToDate(-50)` - Returns 1970-01-01.
- `ToDate(22222222)` - Returns 1970-01-01.
- `ToDate("2004 2 19")` - Returns 2004-2-19.
- `ToDate("99-5")` - Returns 1999-5-1.
- `ToDate("1999-2-29")` - Returns null since 1999 is not leap year.

ToDate(Date a)

This function returns the specified date.

Parameter

a - A Date value.

Return value

The return value is a Date.

Example

Suppose the date is July 15, 1999, the return of the following statement is 07/15/99.

```
ToDate(ToDate(1999, 7, 15))
```

ToDate(Datetime a)

This function extracts the date portion of a specified DateTime.

Parameter

a - A TimeStamp value.

Return value

The return value is a Date value.

Example

Suppose the DateTime is July 15,1999 12:20:30, the return value of the following statement is 07/15/99.

```
ToDate(ToDateTime(1999, 7, 15, 12, 20, 30 ) )
```

ToDate(integer d)

This function converts the argument d (A milliseconds value represents the number of milliseconds that have passed since January 1, 1970 00:00:00.000 GMT) to a data value.

Parameter

d - A Milliseconds value, a DBField value or a parameter. It should be a long integer.

Return value

The return value is a Date.

Example

The return value of the following statement is 01/02/70.

```
ToDate(72000000)
```

Note: When the milliseconds value is larger than 2^{32}, you should use DBField or parameter instead.

ToDate(integer y, Integer m, Integer d)

This function creates a Date value from arguments y, m and d.

Parameters

- y - A BigInt value indicating year.
- m - A BigInt value indicating month.
- d - A BigInt value indicating day.

Return value

The return value is a Date.

Example

The return value of the following statement is 07/15/99.

```
ToDate(1999, 7, 15)
```

ToDateTime()

ToDateTime(number or string)

Use the function IsDateTime to check whether the number/string is right or not. If it is right, the function returns a DateTime value between 100-1-1 and 9999-12-31, otherwise the function returns null.

Parameters

- number - A Number value or expression to be tested for convertibility to a DateTime value. It can be positive, negative or fractional. It is interpreted as a number of days from Jan 1, 1900.
- string - A String value or expression to be tested for convertibility to a DateTime value. Many forms are accepted.

Return value

DateTime value or null.

Examples

- `ToDateTime(255.35)` - Returns 1900-09-12 08:23:59.
- `ToDateTime("Jan 3, 2004")` - Returns 2004-01-03 00:00:00.
- `ToDateTime("5:25:26pm")` - Returns 2004-02-23 17:25:26 because the current date is 2004-2-23.
- `ToDateTime("2003-2-29 15:26:25")` - Returns null because the given String value is not right DateTime value.
- `ToDateTime("2004 *2 *29 15:26:25")` - Returns null because the given String format is not right DateTime format.

ToDateTime(date a)

Returns a DateTime value.

Parameter

a - A Date value that can be converted to a DateTime type value by this function.

Example

`ToDateTime(ToDate("2004-2-23"))` - Returns 2004-02-23 00:00:00.

ToDateTime(Date a, Time b)

This function creates a TimeStamp value from arguments a and b.

Parameters

- a - A Date value.
- b - A Time value.

Return value

The return value is a DateTime.

Example

Suppose the date is July 15,1999 and the time is 12:20:30, the return value of the following statement is 1999-07-15 12:20:30.

```
ToDateTime(ToDate(1999, 7, 15), ToTime(12, 20, 30))
```

ToDateTime(Integer d)

This function converts the argument d (A milliseconds value represents the number of milliseconds that have passed since January 1, 1970 00:00:00.000 GMT) to a TimeStamp value.

Parameter

d - A Milliseconds value, a DBField value or a parameter. It should be a long integer.

Return value

The return value is a DateTime.

Example

The return value of the following statement is 1970-01-02 04:00:00.

```
ToDateTime(72000000)
```

Notes:

- In this function, you can decide the display format by selecting the format from the Report Inspector. The format of the example above is yyyy-mm-dd hh:mm:ss.
- When the milliseconds value is larger than 2^{32}, you should use DBField or parameter instead.

ToDateTime(Integer y, Integer m, Integer d)

This function creates a TimeStamp value from arguments y, m and d.

Parameters

- y - A BigInt value indicating year.
- m - A BigInt value indicating month.
- d - A BigInt value indicating day.

Return value

The return value is a DateTime.

Example

Suppose the date is July 15,1999, the return value of the following statement is 1999-07-15 12:00:00.

```
ToDateTime(1999, 7, 15)
```

Note: In this function, because you only input year, month and day, the system will output 12:00:00 as the default time. Also you can decide the display format by selecting the format from the Report Inspector. The format of the example above is yyyy-mm-dd hh:mm:ss.

ToDateTime(Integer y, Integer m, Integer d, Integer h, Integer i, Integer s)

This function creates a TimeStamp value from arguments y, m, d, h, i and s.

Parameters

- y - A BigInt value indicating year.
- m - A BigInt value indicating month.
- d - A BigInt value indicating day.
- h - A BigInt value indicating hour.
- i - A BigInt value indicating minute.
- s - a BigInt value indicating second.

Return value

The return value is a DateTime.

Example

Suppose the date is July 15,1999 and time is 8:05:30, the return value of the following statement is 1999-07-15 08:05:30.

```
ToDateTime(1999, 7, 15, 8, 5, 30)
```

Today()

This function returns the current date on a report.

Return value

The return value is a date.

Example

If the current date is Oct. 11, 1999, the return value of the following statement is 10/11/99.

```
Today()
```

ToTime()

ToTime(number or string)

Use the function IsTime to check whether the number/string is right or not. If the number/string is right, the function returns a Time value between 00:00:00 and 23:59:59, otherwise it will return null.

Parameters

- number - A Number value or expression to be tested for convertibility to a Time value. It can be positive, negative or fractional. It is interpreted as units of 24 hours.
- string - A String value or expression to be tested for convertibility to a Time value. Many forms are accepted.

Return value

Time value or null.

Examples

- `ToTime(0.2)` - Returns 04:48:00.
- `ToTime(-0.2)` - Returns 19:12:00.
- `ToTime(50)` - Returns 00:00:00.
- `ToTime("10:34:25")` - Returns 10:34:25.
- `ToTime("2:25pm")` - Returns 02:25:00.
- `ToTime("2004-2-20")` - Returns 00:00:00.
- `ToTime("24:25:23")` - Returns 00:25:23.
- `ToTime("20*23:25")` - Returns null since the given String format is not time format.

ToTime(DateTime a)

This function extracts the time portion from a specified TimeStamp value.

Parameter

a - A TimeStamp value.

Return value

The return value is a Time.

Example

Suppose the date is July 15,1999 and time is 10:55:45. The return value of the following statement is 10:55:45.

```
ToTime(ToDateTime(1999, 7, 15, 10, 55, 45 ))
```

ToTime(Integer t)

This function converts the argument t (A milliseconds value represents the number of milliseconds that have passed since January 1, 1970 00:00:00.000 GMT) to a Time value.

Parameter

t - A milliseconds value, a DBField value or a parameter. It should be a long integer.

Return value

The return value is a Time.

Example

The return value of the following statement is 08:00:01.

```
ToTime(1000)
```

Note: When the milliseconds value is larger than 2^{32}, you should use DBField or parameter instead.

ToTime(Integer a, Integer b, Integer c)

This function creates a Time value from argument a, b and c.

Parameters

- a - A BigInt value, indicating hours.
- b - A BigInt value, indicating minutes.
- c - A BigInt value, indicating seconds.

Return value

The return value is a Time.

Example

The return value of the following statement is 10:10:10.

```
ToTime(10, 10, 10)
```

WeekdayName(weekday, abbreviate, firstDayOfWeek)

WeekdayName returns a string indicating the name of the specified day of the week.

Overloads

- WeekdayName(weekday)
- WeekdayName(weekday, abbreviate)
- WeekdayName(weekday, abbreviate, firstDayOfWeek)

Parameters

- Weekday - The numeric designation for the day of the week.
- abbreviate - An optional Boolean value that indicates whether or not the weekday name is to be abbreviated.
- FirstDayOfWeek - An optional number indicating the first day of the week.

Constant	Description
jrUseSystem	0 (Use the current system date)
jrSunday	1
jrMonday	2
jrTuesday	3
jrWednesday	4
jrThursday	5
jrFriday	6
jrSaturday	7

Return value

String value.

Examples

- `WeekdayName (2)` - Returns String value Monday.
- `WeekdayName (1)` - Returns String value Sunday.
- `WeekdayName (4, true)` - Returns String value Wed.
- `WeekdayName (4, false)` - Returns String value Wednesday.
- `WeekdayName (3, true, "jrMonday")` - Returns Wed, since abbreviation is selected and the first day of the week is specified to be Monday.
- `WeekdayName (3, true, "jrUseSystem")` - If the current day is Friday, the function returns Sun, since the first day of week is specified to the current day, and the third day from Friday is Sunday.

WeekFrom1970()

WeekFrom1970()

This function computes the total number of weeks from 1970 to the current date.

Return value

The return value is an Integer.

Example

If the current date is Oct. 13,1999, the return value of the following statement is 1554.

```
WeekFrom1970()
```

WeekFrom1970(Date, a)

This function computes the total number of weeks from 1970 to a specified date.

Parameter

a - A Date value.

Return value

The return value is an Integer.

Example

Suppose the date is July 8,1995. The return value of the following statement is 1331.

```
WeekFrom1970(ToDate(1995, 7, 8))
```

WeekFrom1970(DateTime, a)

This function computes the total number of weeks from 1970 to a specified DateTime.

Parameter

a - A DateTime value.

Return value

The return value is an Integer.

Example

Suppose the DateTime is Jan. 2,1988 10:40:50. The return value of the following statement is 940.

```
WeekFrom1970(ToDateTime(1988, 1, 2, 10, 40, 50))
```

WeekOfMonth()

WeekOfMonth()

This function returns the week of the current month.

Return value

The return value is an Integer.

Example

If the current date is Oct. 13,1999, the return value of the following statement is 3.

```
WeekOfMonth()
```

WeekOfMonth(Date a)

This function returns the week of the month of a specified date.

Parameter

a - A Date value.

Return value

The return value is an Integer.

Example

If the current date is Oct. 13,1999, the return value of the following statement is 3.

```
WeekOfMonth(ToDate(1999, 10, 13))
```

WeekOfMonth(DateTime a)

This function evaluates the week of the month of a specified TimeStamp.

Parameter

a - A TimeStamp value.

Return value

The return value is an Integer.

Example

Suppose the timestamp is Jan. 20,1999 10:40:50. The return value of the following statement is 4.

```
WeekOfMonth(ToDateTime(1999, 1, 20, 10, 40, 50))
```

WeekOfYear()

WeekOfYear()

This function returns the week of the current year.

Return value

The return value is an Integer.

Example

If the current date is Oct. 13,1999, the return value of the following statement is 42.

```
WeekOfYear()
```

WeekOfYear(Date a)

This function returns the week of the year in a specified date.

Parameter

a - A Data value.

Return value

The return value is an Integer.

Example

Suppose the date is July 8,1999. The return value of the following statement is 28.

```
WeekOfYear(ToDate(1999, 7, 8))
```

WeekOfYear(DateTime a)

This function returns the week of the year in a specified TimeStamp.

Parameter

a - A TimeStamp value.

Return value

The return value is an Integer.

Example

Suppose the timestamp is Feb. 4,1999 10:40:50. The return value of the following statement is 6.

```
WeekOfYear(ToDateTime(1999, 2, 4, 10, 40, 50))
```

Year()

Year()

This function returns the current year.

Return value

The return value is an Integer.

Example

If the current year is 1999, the return value of the following statement is 1999.

```
Year()
```

Year(Date a)

This function extracts the year portion of a specified date.

Parameter

a - A Data value.

Return value

The return value is an Integer.

Example

Suppose the date is Jan.15,1995. The return value of the following statement is 1995.

```
Year(ToDate(1995, 1, 15))
```

Year(DateTime a)

This function extracts the year portion of a specified TimeStamp value.

Parameter

a - A TimeStamp value.

Return value

The return value is an Integer.

Example

Suppose the timestamp is July 15,1988 10:40:50. The return value of the following statement is 1988.

```
Year(ToDateTime(1988, 7, 15, 10, 40, 50))
```

Financial

The following are functions of the Financial type.

- DDB()
- FRAccRecTurnover()
- FRCashFlowVsTotalDebt()
- FRCurrentRatio()
- FRDebtEquityRatio()
- FRDividendYield()
- FREarningsPerCommonShare()
- FREquityVsTotalAssets()
- FRGrossProfitMargin()
- FRInterestCoverage()
- FRInventoryTurnover()
- FRNetProfitMargin()
- FROperatingProfitMargin()
- FRPriceEarningsRatio()
- FRQuickRatio()
- FRReturnOnCommonEquity()
- FRReturnOnEquity()
- FRReturnOnInvestedCapital()
- FRReturnOnNetFixedAssets()
- FRReturnOnTotalAssets()
- FV()
- IPmt()
- IRR()
- MIRR()
- NPer()
- NPV()
- Pmt()
- PPmt()
- PV()
- Rate()
- SLN()
- SYD()

DDB(cost, salvage, life, period, factor)

DDB returns a number specifying the depreciation of an asset for a specific time period, using the double-declining balance method or another method as specified by the factor argument.

Overloads

- DDB(cost, salvage, life, period)
- DDB(cost, salvage, life, period, factor)

Parameters

- cost - A Number or Currency that specifies the initial cost of the asset. The value is nonnegative and greater than or equal to salvage.
- salvage - A Number or Currency that specifies the value of the asset at the end of its useful life. The value is nonnegative.
- life - A positive Number that specifies the length of the useful life of the asset.
- period - A Number that specifies the period for which asset depreciation is calculated. The value is positive and less than or equal to life. The parameters life and period must have the same units.
- factor - An optional positive number that specifies the rate at which the balance declines. If omitted, 2 (double-declining method) will be used.

Return value

Number value.

Examples

Suppose a company purchases a fleet of cars for $560,000. The cars have a lifetime of 12 years and a salvage value of $30,000. They are depreciated using the double-declining method.

- `DDB(560000, 30000, 12, 1)` - Returns 93333.33. The first year's depreciation is $93,333.33.
- `DDB(560000, 30000, 13, 5)` - Returns 44164.37. The fourth year's depreciation is $44,164.37.
- `DDB(560000, 30000, 13, 13)` - Returns 11605.58. The final year's depreciation is $11,605.58.

FRAccRecTurnover(Number accountReceivable, Number sales, DbDouble numOfDays)

This function computes the turnover of the account receivable.

Parameters

- accountReceivable - A Double value indicating the account receivable.
- sales - A Double value indicating the sales.
- numOfDays - A Double value indicating the number of days.

Return value

The return value is a Number.

Examples

- If the account receivable is 220000.00, the sales value is 450000.00, the number of days is 100, the return value of the following statement is 48.89.

  `FRAccRecTurnover(220000.00, 450000.00, 100)`
- `FRAccRecTurnover(10000,100000,360)` - Returns 36 days.

FRCashFlowVsTotalDebt(Number cashFlow,Number totalDebt)

This function returns the result of cash flow vs total debt.

Parameters

- cashFlow - A Double value indicating the cash flow.
- totalDebt - A Double value indicating the total debt.

Return value

The return value is a Number.

Example

If the cash flow is 250000.00, the total debt is 280000.00, the return value of the following statement is 0.89.

`FRCashFlowVsTotalDebt(250000.00, 280000.00)`

FRCurrentRatio(Number curAssets, Number curLiabilities)

This function computes the current ratio.

Parameters

- curAssets - A Double value indicating the current ratio.
- curLiabilities - A Double value indicating the current liabilities.

Return value

The return value is a Number.

Example

If the current assets are 1800000.00, the current liabilities are 150000.00, the return value of the following statement is 12.00.

`FRCurrentRatio(1800000.00, 150000.00)`

FRDebtEquityRatio(Number totalLiabilities, Number totalEquity)

This function computes the debt equity ratio.

Parameters

- totalLiabilities - A Double value indicating the total liabilities.
- totalEquity - A Double value indicating the total equity.

Return value

The return value is a Number.

Example

If the total liabilities are 175000.00, the total equity is 215000.00, the return value of the following statement is 0.81.

`FRDebtEquityRatio(175000.00, 215000.00)`

FRDividendYield(Number dividEnd, Number marketPrice)

This function computes the dividend yield.

Parameters

- dividend - A Double value indicating the dividend.
- marketPrice - A Double value indicating the market price.

Return value

The return value is a Number.

Example

If the dividend is 9.85, the market price is 10.87, the return value of the following statement is 0.91.

`FRDividendYield(9.85, 10.87)`

FREarningsPerCommonShare(Number netProfit, Number preferredDividend, DbDouble numOfCommonShare)

This function computes the earnings of per common share.

Parameters

- netProfit - A Double value indicating the net profit.
- preferredDividend - A Double value indicating the preferred dividend.
- numOfCommonShare - A Double value indicating the number of common shares.

Return value

The return value is a Number.

Example

If the net profit is 300000.00, the preferred dividend is 180000.00, the number of common shares is 10000, the return value of the following statement is 12.

`FREarningsPerCommonShare(300000.00, 180000.00, 10000)`

FREquityVsTotalAssets(Number totalEquity, Number totalAssets)

This function returns the result of equity vs. total assets.

Parameters

- totalEquity - A Double value indicating the total equity.
- totalAssets - A Double value indicating the total assets.

Return value

The return value is a Number.

Example

If the total equity is 215000.00, the total assets are 2200000.00, the return value of the following statement is 0.10.

`FREquityVsTotalAssets(215000.00, 2200000.00)`

FRGrossProfitMargin(Number grossProfit, Number sales)

This function is used to compute the gross profit margin.

Parameters

- grossProfit - A Double value indicating the gross profit.
- sales - A Double value indicating the sales.

Return value

The return value is a Number.

Example

If the gross profit is 350000.00, the sales value is 450000.00, the return value of the following statement is 0.78.

`FRGrossProfitMargin(350000.00, 450000.00)`

FRInterestCoverage(number cashFlow, number interestExpenses)

This function computes the interest coverage.

Parameters

- cashFlow - A Double value indicating the cash flow.
- interestExpenses - A Double value indicating the interest expenses.

Return value

The return value is a Number.

Example

If the cash flow is 250000.00, the interest expenses are 350000.00, the return value of the following statement is 0.71.

`FRInterestCoverage(250000.00, 350000.00)`

FRInventoryTurnover(Number inventory, Number sales, Number numOfDays)

This function computes the turnover of the inventory.

Parameters

- inventory - A Double value indicating the inventory.
- sales - A Double value indicating the sales.
- numOfDays - A Double value indicating the number of days.

Return value

The return value is a Number.

Example

If the inventory is 165000.00, the sales value is 450000.00, the number of days is 100, the return value of the following statement is 36.67.

`FRInventoryTurnover(165000.00, 450000.00, 100)`

FRNetProfitMargin(Number netProfit, Number sales)

This function is used to compute the net profit margin.

Parameters

- netProfit - A Double value indicating the net profit.
- sales - A Double value indicating the sales.

Return value

The return value is a Number.

Example

If the net profit is 300000.00, the sales value is 450000.00, the return value of the following statement is 0.67.

`FRNetProfitMargin(300000.00, 450000.00)`

FROperatingProfitMargin(Number operatingProfit, Number sales)

This function computes the operating profit margin.

Parameters

- operatingProfit - A Double value indicating the operating profit.
- sales - A Double value indicating the sales.

Return value

The return value is a Number.

Example

If the operating profit is 380000.00, the sales value is 450000.00, the return value of the following statement is 0.84.

`FROperatingProfitMargin(380000.00, 450000.00)`

FRPriceEarningsRatio(Number marketPrice, Number earningsPerShare)

This function computes the price earning ratio.

Parameters

- marketPrice - A Double value indicating the market price.
- earningPerShare - A Double value indicating the earning per share.

Return value

The return value is a Number.

Example

If the market price is 10.87, the earning per share is 12, the return value of the following statement is 0.91.

`FRPriceEarningsRatio(10.87, 12)`

FRQuickRatio(Number curAssets, Number inventories, Number curLiabilities)

This function computes the quick ratio.

Parameters

- curAssets - A Double value indicating the current assets.
- inventories - A Double value indicating the inventory.
- curLiabilities - A Double value indicating the current liabilities.

Return value

The return value is a Number.

Example

If the current assets are 1800000.00, the inventory is 165000.00, the current liabilities are 150000.00, the return value of the following statement is 10.90.

`FRQuickRatio(1800000.00, 165000.00, 150000.00)`

FRReturnOnCommonEquity(Number netProfit, Number preferredDividend, Number commonEquity)

This function returns the result of return on common equity.

Parameters

- netProfit - A Double value indicating the net profit.
- preferredDividend - A Double value indicating the preferred dividend.
- commonEquity - A Double value indicating the common equity.

Return value

The return value is a Number.

Example

If the net profit is 300000.00, the preferred dividend is 180000.00, the common equity is 208000.00, the return value of the following statement is 0.58.

`FRReturnOnCommonEquity(300000.00, 180000.00, 208000.00)`

FRReturnOnEquity(Number netProfit, Number totalEquity)

This function returns the results of return on equity.

Parameters

- netProfit - A Double value indicating the net profit.
- totalEquity - A Double value indicating the total equity.

Return value

The return value is a Number.

Example

If the net profit is 300000.00, the total equity is 215000.00, the return value of the following statement is 1.40.

`FRReturnOnEquity(300000.00, 215000.00)`

FRReturnOnInvestedCapital(Number netProfit, Number totalBankDebts, Number totalEquity)

This function returns the result of return on invested capital.

Parameters

- netProfit - A Double value indicating the net profit.
- totalBankDebts - A Double value indicating the total bank debts.
- totalEquity - A Double value indicating the total equity.

Return value

The return value is a Number.

Example

If the net profit is 300000.00, the total bank debts are 100000.00, the total equity is 215000.00, the return value of the following statement is 0.95.

`FRReturnOnInvestedCapital(300000.00, 100000.00, 215000.00)`

FRReturnOnNetFixedAssets(Number netProfit, Number netFixedAssets)

This function returns the result of return on net fixed assets.

Parameters

- netProfit - A Double value indicating the net profit.
- netFixedAssets - A Double value indicating the net fixed assets.

Return value

The return value is a Number.

Example

If the net profit is 300000.00, the net fixed assets are 400000.00, the return value of the following statement is 0.75.

`FRReturnOnNetFixedAssets(300000.00, 400000.00)`

FRReturnOnTotalAssets(Number netProfit, Number totalAssets)

This function returns the results of return on total assets.

Parameters

- netProfit - A Double value indicating the net profit.
- totalAssets - A Double value indicating the total assets.

Return value

The return value is a Number.

Example

If the net profit is 300000.00, the total assets are 2200000.00, the return value of the following statement is 0.14.

`FRReturnOntotalAssets(300000.00, 2200000.00)`

FV(rate, periods, payment, presentMoney, type)

Returns a number specifying the future value of an annuity based on periodic, fixed payments and a fixed interest rate.

Overloads

- FV(rate, periods, payment)
- FV(rate, periods, payment, presentMoney)
- FV(rate, periods, payment, presentMoney, type)

Parameters

- rate - A Number that specifies the interest rate per period.
- periods - A positive Number that specifies the total number of payment periods in the annuity. The units used for specifying rate and periods must consistent. For example, if periods is the number of periods in months, then rate is a monthly interest rate.
- payment - A Number or Currency that specifies the payment to be made each period.
- presentMoney - An optional Number or Currency that specifies the present value of a series of future payments.
- type - An optional Number that specifies when payments are due. Specify 0 if payments are due at the end of the payment period, and 1 if payments are due at the beginning of the period. If omitted, 1 will be used.

Return value

Number value.

Examples

- Suppose that you put $1100 per month into a retirement savings plan that pays 6 percent annual interest, compounded monthly. How much will the account be worth after 20 years?

  `FV(0.06 / 12, 20 * 12, -1100)` - Returns 507144.99. So your account will have $507,145. The payment (-1100) is negative since you are paying out the money to the plan.

  The above example assumes that you make your payments into the plan at the end of the month. Thus, after the first month, your plan would have $1100 in it, since there wouldn't have been enough time for any interest to accrue.
- Suppose that instead, you make your payments at the start of the month,

  `FV(0.06 / 12, 20 * 12, -1100, 0, 1)` - Returns 510786.21 (rounded to the nearest integer). So your account will have $510,786. You will save $3,642 more by depositing at the beginning of the month.
- Now suppose that in addition to making payments at the start of the month, you start your plan with an initial deposit of $25,000,

  `FV(0.06 / 12, 20 * 12, -1100, -25000, 1)` - Returns 593541.33 (rounded to the nearest integer). Your account will have $ 593,541 after 20 years.
- You can also use the FV function to calculate the future value of a lump sum deposit. For example, if you deposit $25,000 into a plan that pays 6 percent annual interest compounded monthly for 20 years,

  `FV(0.06 / 12, 20 * 12, 0, -25000)` - Returns 82755.12 (rounded to the nearest integer). You would have $82755.12 in your account.

IPmt(rate, period, periods, presentMoney, futureMoney, type)

IPmt returns a number specifying the interest payment for a given period of an annuity based on periodic, fixed payments and a fixed interest rate.

Overloads

- IPmt(rate, period, periods, presentMoney)
- IPmt(rate, period, periods, presentMoney, futureMoney)
- IPmt(rate, period, periods, presentMoney, futureMoney, type)

Parameters

- rate - A Number that specifies the interest rate per period.
- period - A Number that specifies the payment period in the range 1 through periods.
- periods - A positive Number that specifies the total number of payment periods in the annuity. The units used for specifying rate, period and periods must be consistent. For example, if periods is the number of periods in months, then rate is a monthly interest rate and period specifies a month.
- presentMoney - A Number or Currency that specifies the present value, or value today, of a series of future payments or receipts.
- futureMoney - An optional Number or Currency that specifies the future value, or cash balance you want after you've made the final payment. If omitted, 0 will be used.
- type - An optional Number that specifies when payments are due. Specify 0 if payments are due at the end of the payment period, and 1 if payments are due at the beginning of the period. If omitted, 0 will be used.

Return value

Number value.

Examples

- Suppose that you want to take out a $250,000 loan payable monthly over 15 years at an annual interest rate of 6.5 percent. The following formula returns the amount of interest that you pay in your first loan payment. Note that the monthly interest rate is 0.065 / 12 and the number of months of the loan is 15 * 12.

  `IPmt(0.065 / 12, 1, 15 * 12, 250000)` - Returns the Number value -1354.17 (rounded to 2 decimals). The value is negative because it represents a payment out from you, whereas the loan amount of $250,000 is positive because it represents a payment in to you.
- The following formula returns the amount of interest that you pay in your 97th payment (after 10 years of payments).

  `IPmt(0.065 / 12, 9*12 + 1, 15 * 12, 250000)` - Returns -701.74 (rounded to 2 decimals). This shows that you would be making progress on the loan at this stage, with less of your monthly payment paying the interest.

IRR(values, guess)

IRR returns a number specifying the internal rate of return for a series of periodic cash flows (payments and receipts).

Overloads

- IRR(values)
- IRR(values, guess)

Parameters

- values - A Number or Currency type array that specifies cash flow values. The array must contain at least one negative value (a payment) and one positive value (a receipt). The cash flows must occur at regular intervals such as monthly or yearly.
- guess - An optional Number value that is estimated to be returned by IRR. If omitted, guess is 0.1 (10 percent).

Return value

Number value.

Example

Suppose that you can choose one of two offers: $20,000 now or guaranteed payments of $5,000 after 1 year, $10,000 after 2 years and $15,000 after 3 years. Which is the better offer? One way to quantify this is to calculate the internal rate of return. If you take the second offer, you cannot take the first, which is like experiencing an initial payment of $20,000 followed by the receipts:

```
inArray = ["-20000", "15000", "-10000", "25000"];
IRR(inArray)
```

Returns 0.201 (rounded to 3 decimals) or 20.1 percent interest. With all other things being equal, if you think that 20.1 percent is a good rate of return, you would prefer the second offer.

Note: The NPV and IRR functions are related since NPV (IRR (values), values) = 0. That is, the internal rate of return of a sequence of cash flows is the interest rate for which that sequence of cash flows has a net present value of 0. There is no direct formula for the IRR function and so JReport calculates the value by iteration. The process depends on the initial guess for the internal rate of return. If the program reports an error, try changing the value of the guess argument to be closer to what you expect the internal rate of return to be.

MIRR(valueArray, financeRate, reinvestRate)

Returns a number specifying the modified internal rate of return for a series of periodic cash flows (payments and receipts).

Parameters

- valueArray - A Number or Currency type array that specifies cash flow values. The array must contain at least one negative value (a payment) and one positive value (a receipt). The cash flows must occur at regular intervals such as monthly or yearly.
- financeRate - A Number that specifies the interest rate paid as the cost of financing.
- reinvestRate - A Number that specifies the interest rate received on gains from cash reinvestment.

Return value

Number value.

Example

Suppose that you run a business that makes equipment investments, which results in a loss in the first and fourth years. Your expected annual returns are: -$60,000, $60,000, $45,000, -$50,000, $65,000, $40,000. Your losses are financed at 10 percent while you reinvest your earnings in an account at 6 percent. The modified internal rate of return is:

`MIRR([-60000, 60000, 45000, -50000, 65000, 40000], 0.10, 0.06)` - Returns 0.1929 (rounded to 4 decimals) or 19.29 percent.

NPer(rate, payment, presentMoney, futureMoney, type)

NPer returns a number specifying the number of periods for an annuity based on periodic, fixed payments and a fixed interest rate.

Overloads

- NPer (rate, payment, presentMoney)
- NPer (rate, payment, presentMoney, futureMoney)
- NPer (rate, payment, presentMoney, futureMoney, type)

Parameters

- rate - A Number that specifies the interest rate per period.
- payment - A Number or Currency that specifies the payment to be made each period. Payments usually contain principal and interest that doesn't change over the life of the annuity.
- presentMoney - A Number or Currency that specifies the present value, or value today, of a series of future payments or receipts.
- futureMoney - An optional number or Currency that specifies the future value or cash balance you want after you've made the final payment. If omitted, 0 will be used.
- type - An optional cumber that specifies when payments are due. Specify 0 if payments are due at the end of the payment period, and 1 if payments are due at the beginning of the period. If omitted, 0 will be used.

Return value

Number value.

Examples

- Suppose that you want a $250,000 loan to buy a house. The interest rate is 6.5 percent and you can afford to pay $3,000 per month. How long a mortgage would you need?

  `NPer(0.075/12, -3000, 250000)` - Returns 118.08 (rounded to 2 decimals) months.
- Rather than seeking a loan, suppose that you want to save $500,000 and then buy the house. You can get a 6.5 percent interest rate compounded monthly from your savings plan, and you want to save $3,000 per month. How long would you need to save before you had the required amount of money?

  `NPer(0.065/12, -3000, 0, 250000)` - Returns 68.96 (rounded to 2 decimals) months.

NPV(rate, values)

Returns a number specifying the net present value of an investment based on a series of periodic cash flows (payments and receipts) and a discount rate.

Parameters

- rate - A Number that specifies the discount rate over the length of the period, expressed as a decimal.
- value - A Number or Currency type array that specifies cash flow values. Negative values represent payments and positive values receipts. The cash flow must occur at regular intervals, such as monthly or yearly.

Return value

Number value.

Example

Suppose that someone offers to pay you $15000 after 1 year, $20000 after 2 years, $25000 after 3 years and $12000 after 4 years. If the discount rate (the Time value of money) is 6.5 percent, the value of this offer to you today is:

`NPV(0.065, [15000, 20000, 25000, 12000])`

The formula returns 61741.8. So this scheme is worth $61,741.8 to you today. This is less than the sum of the payments, which is $72000, since you have to wait for this money.

Note: The NPV and IRR functions are related since NPV (IRR (values), values) = 0, which means that the internal rate of the return of a sequence of cash flow is the discount rate for which that sequence of cash flow has a net present value of 0.

Pmt(rate, periods, presentMoney, futureMoney, type)

Returns a number specifying the payment for an annuity based on periodic, fixed payments and a fixed interest rate. To find the total amount paid out over the whole loan, multiply the payment per period (the value returned by Pmt) by the total number of periods.

Overloads

- Pmt(rate, periods, presentMoney)
- Pmt(rate, periods, presentMoney, futureMoney)
- Pmt(rate, periods, presentMoney, futureMoney, type)

Parameters

- rate - A Number that specifies the interest rate per period.
- periods - A positive Number that specifies the total number of payment periods in the annuity. The units used for specifying rate and periods must be consistent. For example, if periods is the number of periods in months, then rate will be a monthly interest rate.
- presentMoney - A Number or Currency that specifies the present value or principal. That is, the amount that a series of payments in the future is worth now.
- futureMoney - An optional Number or Currency that specifies the future value or cash balance you want after you've made the final payment. If omitted, 0 will be used.
- type - An optional Number that specifies when payments are due. Specify 0 if payments are due at the end of the payment period, and 1 if payments are due at the beginning of the period. If omitted, 0 will be used.

Return value

Number value.

Examples

- Suppose that you want to take out a $250,000 loan payable monthly over 15 years at an annual interest rate of 6.5 percent. The following formula returns your monthly loan payment. Note that the monthly interest rate is 0.065 / 12, and the number of months of the loan is 15 * 12.

  `Pmt(0.065 / 12, 15 * 12, 250000)` - Returns the Number value -2177.77 (rounded to 2 decimals). The value is negative because it represents a payment out from you whereas the loan amount of $250,000 is positive because it represents a payment in to you.
- Now suppose that the payments are made at the beginning of the month instead of the end (default). Your monthly loan payment is calculated as:

  `Pmt(0.065 / 12, 15 * 12, 250000, 0, 1)` - Returns -2166.04 (rounded to 2 decimals). Note that your monthly payment is about $11.73 less each month than in the previous example where payments are made at the end of the month.
- Now suppose that you know that you'll receive $80,000 in 15 years. In this case, there is no need to fully pay off the loan. You only need to reduce the amount owed to $80,000 after 15 years. Note that the future value is negative since after 15 years you will need to pay out $80,000 to clear the loan. Your monthly loan payment is calculated as:

  `Pmt(0.065 / 12, 15 * 12, 250000, -80000)` - Returns -1914.22 (rounded to 2 decimals).

PPmt(rate, period, periods, presentMoney, futureMoney, type)

Returns a number specifying the principal payment for a given period of an annuity based on periodic, fixed payments and a fixed interest rate.

Overloads

- PPmt(rate, period, periods, presentMoney)
- PPmt(rate, period, periods, presentMoney, futureMoney)
- PPmt(rate, period, periods, presentMoney, futureMoney, type)

Parameters

- rate - A Number that specifies the interest rate per period.
- period - A Number that specifies the payment period in the range 1 through periods.
- periods - A positive Number that specifies the total number of payment periods in the annuity. The units used for specifying rate, period and periods must be consistent. For example, if periods is the number of periods in months, then rate is a monthly interest rate and period specifies a month.
- presentMoney - A Number or Currency that specifies the present value, or value today, of a series of future payments or receipts.
- futureMoney - An optional Number or Currency that specifies the future value or cash balance you want after you've made the final payment. If omitted, 0 will be used.
- type - An optional Number that specifies when payments are due. Specify 0 if payments are due at the end of the payment period, and 1 if payments are due at the beginning of the period. If omitted, 0 will be used.

Return value

Number value.

Examples

- Suppose that you want to take out a $250,000 loan payable monthly over 15 years at an annual interest rate of 6.5 percent. The following formula returns the amount of principal that you pay in your first loan payment. Note that the monthly interest rate is 0.065 / 12 and the number of months of the loan is 15 * 12.

  `PPmt (0.065 / 12, 1, 15 * 12, 250000)` - Returns the Number value -823.6 (rounded to 2 decimals). The value is negative because it represents a payment out from you whereas the loan amount of $250,000 is positive because it represents a payment in to you.
- The following formula returns the amount of principal that you pay in your 121st payment (after 10 years of payments):

  `PPmt (0.065 / 12, 9*12 + 1, 15 * 12, 250000)` - Returns -1476.03 (rounded to 2 decimals).

  You've made progress on the loan and are paying off more of the principal per payment. This is because less interest can accrue each month since more of the loan is paid off so that your fixed monthly payment is applied more to the principal.

PV(rate, periods, payment, futureMoney, type)

Returns a number specifying the present value of an annuity based on periodic, fixed payments to be paid in the future and at a fixed interest rate.

Overloads

- PV (rate, periods, payment)
- PV (rate, periods, payment, futureMoney)
- PV (rate, periods, payment, futureMoney, type)

Parameters

- rate - A Number that specifies the interest rate per period.
- periods - A positive Number that specifies the total number of payment periods in the annuity. The units used for specifying rate and periods must be consistent. For example, if periods is the number of periods in months, rate will then be a monthly interest rate.
- payment - A Number or Currency that specifies payment to be made each period.
- futureMoney - an optional Number or Currency that specifies the future value or cash balance you want after you've made the final payment. If omitted, 0 will be used.
- type - An optional Number that specifies when payments are due. Specify 0 if payments are due at the end of the payment period, and 1 if payments are due at the beginning of the period. If omitted, 0 will be used.

Return value

Number value.

Example

Suppose that you want to buy a condo and can make payments of $1100 twice a month (24 annual payments). If the mortgage rates are 6.5 percent, and you want to pay off the condo in 10 years, what is the maximum loan that you can take out?

`PV(0.065 / 24, 10 * 24, -1100)` - Returns 193936 (rounded to the nearest dollar).

You can therefore afford a loan of about $194,000. Notice that the payment argument is negative since you are paying out the money each month.

Rate(periods, payment, presentMoney, futureMoney, type, guess)

Rate returns a number specifying the interest rate per period for an annuity. The units of the returned value are consistent with the units of periods. For example, if periods is in months, then the rate returned will be a monthly interest rate.

Overloads

- Rate(periods, payment, presentMoney)
- Rate(periods, payment, presentMoney, futureMoney)
- Rate(periods, payment, presentMoney, futureMoney, type)
- Rate(periods, payment, presentMoney, futureMoney, type, guess)

Parameters

- periods - A positive Number that specifies the total number of payment periods in the annuity.
- payment - A Number or Currency that specifies the payment that is to be made for each period.
- presentMoney - A Number or Currency that specifies the present value, or value today, of a series of future payments or receipts.
- futureMoney - An optional Number or Currency that specifies the future value or cash balance you want after you've made the final payment. If omitted, 0 will be used.
- type - An optional Number that specifies when payments are due. Specify 0 if payments are due at the end of the payment period, and 1 if payments are due at the beginning of the period. If omitted, 0 will be used.
- guess - An optional Number value estimated to be returned by Rate. If omitted, guess is 0.1 (10 percent).

Return value

Number value.

Examples

- An electronics store offers to finance a $12500 television for $560 per month, over 2 years, with no money down. Is this a good deal? The first step in determining this is to figure out what interest rate the store is charging.

  `Rate(2 * 12, -560, 12500)` - Returns 0.00588 (rounded to 5 decimals). Notice that periods is 24 months, payment (-560) is negative since monthly payments are being paid, and the present value (12500) is positive since at the start of the loan, $12500 was effectively received (the value of the television). The interest rate returned is a monthly interest rate, since periods is in months.
- This next expression calculates the interest rate, expressed as a yearly interest rate and as a percent.

  `Rate(2 * 12, -560, 12500) * 12 * 100` - Returns 7.06 (rounded to 2 decimals). Thus, the store is charging an effective annual interest rate of 7.06 percent.

Note: There is no direct formula for the Rate function and so Reports calculates the value by iteration. The process depends on the initial guess for the rate. If the program reports an error, try changing the value of the guess argument to be closer to what you expect the interest rate to be.

SLN(cost, salvage, life)

SLN returns a number specifying the straight-line depreciation of an asset for a single period.

Parameters

- cost - A Number or Currency that specifies the initial cost of the asset.
- salvage - A Number or Currency that specifies the value of the asset at the end of its useful life.
- life - A Number that specifies the length of the useful life of the asset. It must not equal 0.

Return value

Number value.

Example

Suppose a company purchases a fleet of cars for $560,000. The cars have a lifetime of 12 years and a salvage value of $30,000. The depreciation per year is:

`SLN(560000, 30000, 12)` - Returns 44166.67. Thus the depreciation per year is $44,166.67.

SYD(cost, salvage, life, period)

SYD returns a number specifying the sum-of-years' digits depreciation of an asset for a single period.

Parameters

- cost - A Number or Currency that specifies the initial cost of the asset.
- salvage - A Number or Currency that specifies the value of the asset at the end of its useful life.
- life - A positive Number that specifies the length of the useful life of the asset.
- period - A Number that specifies the period for which asset depreciation is calculated. The value is positive and less than or equal to life. The Parameters life and period must have the same units.

Return value

Number value.

Examples

Suppose a company purchases a fleet of cars for $560,000. The cars have a lifetime of 12 years and a salvage value of $30,000. They are depreciated as follows:

- `SYD(560000, 30000, 12, 1)` - Returns 80000. The first year's depreciation is $81,538.46.
- `SYD(560000, 60000, 12, 5)` - Returns 56000. The fourth year's depreciation is $54,358.97.
- `SYD(560000, 60000, 12, 12)` - Returns 8000. The final year's depreciation is $6,794.87.

Math

Mathematical functions are used for a variety of mathematics-oriented calculations and operations.

- Abs()
- atan()
- Average()
- cos()
- Count()
- DistinctCount()
- e()
- exp()
- integDiv()
- log()
- Maximum()
- Minimum()
- MRound()
- pi()
- PopulationStdDev()
- PopulationVariance()
- pow()
- random()
- Remainder()
- Round()
- RunningAvg()
- RunningCount()
- RunningDistinctCount()
- RunningMaximum()
- RunningMinimum()
- RunningSum()
- sgn()
- sin()
- sqr()
- StdDev()
- Sum()
- tan()
- ToBinary()
- ToHex()
- ToInt()
- ToOctal()
- Truncate()
- Variance()

Abs()

Abs(BigInt, a)

Returns the absolute value of a BigInt value. If the argument is not negative, the argument will be returned. If the argument is negative, the negation of the argument will be returned.

Parameter

a - A BigInt value.

Return value

The return value is an Integer.

Example

The return value of the following statements is 3534.

`Abs(-3534) and Abs(3534)`

Abs(currency, a)

Returns the absolute value of a Currency value. If the argument is not negative, the argument will be returned. If the argument is negative, the negation of the argument will be returned.

Parameter

a - A Currency value.

Return value

The return value is a Currency.

Example

The return value of the following statement is $3.67.

`Abs($3.67) and Abs($-3.67)`

Abs(double, a)

Returns the absolute value of a Double value. If the argument is not negative, the argument will be returned. If the argument is negative, the negation of the argument will be returned.

Parameter

a - A Double value.

Return value

The return value is a Number.

Example

The return value of the following statements is 2.4785.

`Abs(2.4785) and Abs(-2.4785)`

atan(number)

Returns a number specifying the arctangent of the given number parameter. In other words, it returns the angle whose tangent is the given number parameter. This function works in a similar way to the Java Math function with the same name. The range of values returned by atan is between -pi/2 and pi/2 radians.

Parameter

number - A Number value.

Return value

Number value, which is an angle specified in number of radians.

Example

`atan(1)` - Returns an angle of 0.7854 radians (rounded to 4 radians). To convert this angle to degrees multiply by 180 / pi. For instance, atan (1) * 180 / pi is 45 degrees.

Average()

Average(DBField a)

The function is used to get the average value of the values referred to by the argument.

Parameter

a - Values of DBField or formula field. Can be Number or Currency data type.

Return value

The return value can be Number or Currency.

Examples

- `Number x = Average(@dbfield)`
- `Number x = Average(@formula)`
- `Currency x = Average(@dbfield)`
- `Currency x = Average(@formula)`

Average(DBField a, String groupby)

This function is used to get the average value of values referred to by the argument.

Parameters

- a - Values of DBField or formula field. Can be Number or Currency data type.
- groupby - A constant string to indicate the groupby field name or a field variable grouped by or a parameter variable grouped by.

Return value

The result value can be Number or Currency, and is depent on the data type of the first argument respectively.

Examples

- `Number x = Average(@dbfield, @dbfield_groupby)`
- `Number x = Average(@dbfield, @formula_groupby)`
- `Number x = Average(@dbfield, @parameter_groupby)`
- `Currency x = Average(@formula, "group_field")`
- `Currency x = Average(@formula, @formula_groupby)`
- `Currency x = Average(@formula, @parameter_groupby)`

Average(Number a[])

This function computes the average of all the values in the array.

Parameter

a - A Number or Currency array.

Return value

The return value is a Number or a Currency.

Examples

- The return value of the following statement is 13.5.
  `Number x = Average([ 11,12,13,14,15,16 ])`
- The return value of the following statement is -13.5.
  `Number x = Average([ -11, -12, -13, -14, -15, -16 ])`
- The return value of the following statement is 11.265.
  `Number x = Average([ 11.25,11.26,11.27,11.28 ])`
- The return value of the following statement is -11.265.
  `Number x = Average([ -11.25, -11.26, -11.27, -11.28 ])`
- The return value of the following statement is 5.75.
  `Currency x = Average([ $5.6, $5.7, $5.8, $5.9 ])`
- The return value of the following statement is -5.75.
  `Currency x = Average([ -$5.6, -$5.7, -$5.8, -$5.9 ])`

Note: If a field is assigned to a variable, for example x, the variable x will lose the characteristic of representing a group of values. The following formula will then be treated as incorrect.

```
Number x = @dbfield;
Number y = Average(x); // system will prompt you that there is no such kind of function.
```

cos(number)

Returns a number specifying the cosine of an angle given in radians. It takes a right-angle triangle, and returns the length of the side adjacent to the specified angle divided by the length of the hypotenuse.

Parameter

number - An angle in number of radians.

Return value

Number value between 1 and -1.

Examples

- `cos(1)` - Returns 0.54 (rounded to 2 decimals). This is the cosine of 1 radian.
- `cos(60 * pi / 180)` - Returns 0.5. This is the cosine of 60 degrees. Before taking the cosine, the angle is converted to radians by multiplying by pi / 180.

Count()

Count(Array a[ ])

This function computes the number of the values in an array.

Parameter

a - A Double array.

Return value

The return value is an Integer.

Examples

- `Count([11.2, 15.4, 13.7, 16.9, 15.2, 14.8, 10.6])` - Returns 7.
- `Count([-20.4, -26.4, -68.7, -84.1])` - Returns 4.
- `Count(["George","Paul","George","John","Ringo","John"])` - Returns 6.
- `Count([11, 15, 13, 16, 15, 14, 10])` - Returns 7.
- `Count([$1, $2, $3.0, $4.5, $45, $67])` - Returns 6.

Count(DBField a)

The function is used to count the number of values referred to by the argument.

Parameter

a - Values of DBField or formula field.

Return value

Returns a Number type value.

Examples

- `Number x = Count(@dbfield)`
- `Number x = Count(@formula)`

Count(DBField a, String groupby)

The function is used to count the number of values referred to by the first argument.

Parameters

- a - Values of DBField or formula field.
- groupby - A constant string to indicate the groupby DBField name, formula name grouped by or a parameter name grouped by.

Return value

Returns a Number type value.

Examples

- `Number y = Count(@"Customer Name", @Customers_Region)`
- `Number x = Count(@formula, @formula_groupby)`
- `Number x = Count(@dbfield, @parametger_groupby)`

Note: If a field is assigned to a variable, for example x, the variable x will lose the characteristic of representing a group of values. The following formula will then be treated as incorrect.

```
Number x = @dbfield;
Number y = Count(x); // system will prompt you that there is no such kind of function.
```

DistinctCount()

DistinctCount(DBField a)

This function computes the number of distinct values referred to by a DBField.

Parameter

a - Values of a DBField or a formula field.

Return value

The return value is an Integer.

Example

If you build a report about customer order, the return value of the following statement is 19.

`DistinctCount(@"Order Date")`

DistinctCount(DBField a, String groupby)

This function is used to count the number of distinct values referred to by the first argument.

Parameters

- a - Values of a DBField or a formula field.
- groupby - A constant string to indicate the DBField name, formula name or parameter name grouped by.

Return value

Returns a Number type value.

Examples

- `Number x = DistinctCount(@dbfield, @dbfield1)`
- `Number x = DistinctCount(@formula, @formula_groupby)`
- `Number x = DistinctCount(@dbfield, @parameter_groupby)`

DistinctCount(Number a[ ])

This function computes the number of distinct values in an array.

Parameter

a - A Number array.

Return value

The return value is an Integer.

Example

The return value of the following statement is 4.

`DistinctCount([1.45, 2.63, 38.1, 1.45, 23.9])`

Note: If a field is assigned to a variable, for example x, the variable x will lose the characteristic of representing a group of values. The following formula will be treated as incorrect.

```
Number x = @dbfield;
Number y = DistinctCount(x); // system will prompt you that there is no such kind of function.
```

e()

Returns the Mathematical value of e, which is 2.7182818 (if rounded to 7 decimal places).

exp(Number)

Returns a number specifying e (the base of natural logarithms) raised to a power. The value of e is approximately 2.7182818284590452354. This function works like the Java Math function with the same name. Numeric overflow occurs if the given number parameter is larger than approximately 705.

Parameter

inNumber - Number value that specifies a power.

Return value

A number which is the value of e ^ Number.

Examples

- `exp(1.5)` - Returns the exponentiation of 1.5, which is approximately 4.4817.
- `exp(10)` - Returns 22,026.47.

integDiv(numbera, numberb)

Returns the integer portion of the value numbera/numberb.

Parameters

- numbera - A Number value.
- numberb - A Number value.

Return value

Whole Number value.

Example

`integDiv(33, 2)` - Returns 16.

log(number)

Returns a number specifying the natural logarithm of a given number. The natural logarithm is the logarithm to the base e, where e is approximately 2.7182818284590452354. If the value of number is less than 0, the function will return NaN.

Parameter

number - The Number value that you want to calculate on.

Return value

The number which is the value of ln(Number).

Examples

- `log(1.5)` - Returns 0.406 (rounded to 3 decimals).
- `log(22026.47)` - Returns 10.

Maximum()

Maximum(a[ ])

This function returns the highest value in an array.

Parameter

a - An array, data type can be Number.

Return value

Returns a Number value.

Example

- `Maximum([ 22.5, 75.81, 236.47, 56.31, 235.78 ])`

Maximum(DBField a)

This function is used to pick up the maximum value from the values referred to by the argument.

Parameter

a - Values of a DBField or a formula field.

Return value

The returned value type is dependent on the data type of the parameter. It may be Number, Currency, String, Date, Time, DateTime or Boolean.

Examples

- `Number x = Maximum(@dbfield)`
- `Date x = Maximum(@formula)`
- `Currency x = Maximum(@dbfield)`
- `Boolean x = Maximum(@formula)`

Maximum(DBField a, String groupby)

This function is used to pick up the maximum value from the values referred to by the first argument.

Parameters

- a - Values of a DBField or of a formula field.
- groupby - A constant string to indicate the DBField name, formula or parameter grouped by.

Return value

The returned type is dependent on the data type of the first argument. It may be Number, Currency, String, Date, Time, DateTime or Boolean.

Examples

- `Time x = Maximum(@dbfield, @dbfield1)`
- `DateTime x = Maximum(@dbfield, @formula_groupby)`
- `String x = Maximum(@dbfield, @parameter_groupby)`
- `Date x = Maximum(@formula, @formula_groupby)`
- `Currency x = Maximum(@formula, @dbfield1)`
- `Currency x = Maximum(@formula, @parameter)`

Note: If a field is assigned to another variable, for example x, the variable x will lose the characteristic of representing a group of values. The following formula will then be treated as incorrect.

```
Number x = @dbfield;
Number y = Maximum(x); // system will prompt you that there is no such kind of function.
```

Minimum()

Minimum(a[ ])

This function returns the lowest value in an array.

Parameter

a - An array which type can be Number.

Return value

Returns a Number value.

Examples

- `Number x = Minimum([ 2.36, 0.12, 1.25, 0.25 ])`
- `Number x = Minimum([1.0, 2.0, 3.0, 4.5, 45.0, 67.0])`
- `Number x[6] = [1.0, 2.0, 3.0, 4.5, 45.0, 67.0];`
  `Number y = Minimum(x)`

Minimum(DBField a)

This function is used to pick up the minimum value from the values referred to by the argument.

Parameter

a - Values of a DBField or a formula field.

Return value

The returned value type is dependent on the data type of the parameter. It can be Number, Currency, String, Date, Time, DateTime or Boolean.

Examples

- `Number x = Minimum(@dbfield)`
- `Date x = Minimum(@formula)`
- `Currency x = Minimum(@dbfield)`
- `Boolean x = Minimum(@formula)`

Minimum(DBfield a, String groupby)

This function is used to pick up the minimum value from the values referred to by the first argument.

Parameters

- a - Values of a DBField or a formula field.
- groupby - A constant string to indicate the DBField name, formula or parameter grouped by.

Return value

The returned value type is dependent on the first argument. It may be Number, Currency, String, Date, Time, DateTime or Boolean.

Examples

- `Time x = Minimum(@dbfield, @dbfield_groupby)`
- `DateTime x = Minimum(@dbfield, @dbfield_groupby)`
- `String x = Minimum(@dbfield, @parameter_groupby)`
- `Date x = Minimum(@formula, @formula_groupby)`
- `Currency x = Minimum(@formula, @dbfield_groupby)`
- `Currency x = Minimum(@formula, @parameter)`

Note: If a field is assigned to a variable, for example x, the variable x will lose the characteristic of representing a group of values. The following formula will then be treated as incorrect.

```
Number x = @dbfield;
Number y = Minimum(x); // system will prompt you that there is no such kind of function.
```

MRound()

Returns a number that is a multiple of the second parameter and at the same time is nearest to the first parameter. This function rounds up (away from zero) if the remainder of the division is greater than or equal to half of the value of the second parameter.

Overloads

MRound(Number,Number)

MRound(Currency,Number)

MRound(Currency,Currency)

Parameters

- Number - A Number value.
- Currency - A Currency value.

Note: Regardless of the positive/negative sign of the second parameter, the result of the function takes the positive/negative sign of the first parameter.

Return value

A number.

Expression

Take MRound(Currency,Number) for an example:

MRound(Currency,Number) = Number*(Integer(Currency/Number) + (1 if Remainder(Currency/Number) >=Number/2, otherwise 0)

Examples

- MRound(11.0,4.0) - Return 12.0.
- MRound($11.0,-4.0) - Return 12.0.
- MRound($-7.0,$2.0) - Return -8.0.

pi()

Returns the Mathematical value of pi, which is 3.1415926 (if rounded to 7 decimal places).

PopulationStdDev()

- PopulationStdDev(a[ ])
- PopulationStdDev(DBField field)
- PopulationStdDev(DBField field, String groupby)

This function computes the population standard deviation of the values referred to by the argument.

Parameters

- a - A Number array.
- field - Values of a DBField or formula field.
- groupby - A constant string to indicate the DBField, formula or parameter grouped by.

Return value

Returns a Number type value.

Examples

- `Number x = PopulationStdDev([1.0, 2.0, 3.0, 4.5, 45.0, 67.0])`
- `Number x[6] = [1.0, 2.0, 3.0, 4.5, 45.0, 67.0];`
  `Number y = PopulationStdDev( x)`
- `Number x = PopulationStdDev(@dbfield)`
- `Number x = PopulationStdDev(@formula)`
- `Number x = PopulationStdDev(@dbfield, @dbfield1)`
- `Number x = PopulationStdDev(@dbfield, @"group_field")`
- `Number x = PopulationStdDev(@dbfield, @parameter)`

Note: If a field is assigned to a variable, for example x, the variable x will lose the characteristic of representing a group of values. Therefore, the following formula will be treated as incorrect.

```
Number x = @dbfield;
Number y = PopulationStdDev(x); // system will prompt you that there is no such kind of function.
```

PopulationVariance()

- PopulationVariance(a[ ])
- PopulationVariance(DBField field)
- PopulationVariance(DBField field, String groupby)

This function computes the population variance of the values referred to by the argument.

Parameters

- a - A Number array.
- field - Values of a DBField or formula field.
- groupby - A constant string to indicate the DBField, formula or parameter grouped by.

Return value

Returns a Number type value.

Examples

- `PopulationVariance([ 2.5, 4.75, 2.36, 1.25 ])`
- `Number x = PopulationVariance([1.0, 2.0, 3.0, 4.5, 45.0, 67.0])`
- `Number x[6] = [1.0, 2.0, 3.0, 4.5, 45.0, 67.0];`
  `Number y = PopulationVariance(x)`
- `Number x = PopulationVariance(@dbfield)`
- `Number x = PopulationVariance(@formula)`
- `Number x = PopulationVariance(@dbfield, @dbfield_groupby)`
- `Number x = PopulationVariance(@dbfield, @formula_groupby)`
- `Number x = PopulationVariance(@dbfield, @parameter)`

Note: If a field is assigned to a variable, for example x, the variable x will lose the characteristic of representing a group of values. Therefore, the following formula will be treated as incorrect.

```
Number x = @dbfield;
Number y = PopulationVariance(x); // system will prompt you that there is no such kind of function.
```

pow(numbera, numberb)

Returns the value of numbera ^ numberb.

Paremeters

- numbera - A Number value.
- numberb - A Number value.

Examples

- `pow(2, 4)` - Returns 16.
- `pow(6.2, 5.1)` - Returns 10995.060323919315.

random(), random(seed)

Returns a random number greater than or equal to 0 and less than 1. If seed is equal to 0, Random returns the random number that was returned from the previous call to random. If seed is not supplied or is greater than 0, then Random will return the next random number in the internally generated sequence of random numbers. If seed is less than 0, then Random will use this value of seed to start a new random number sequence and returns the first value in the sequence. This function is typically used when your formula requires a randomly generated number, for example, statistical calculations or selecting records at random to limit the data in a report.

Overloads

- random()
- random(seed)

Parameter

seed - An optional Number value parameter.

Return value

A Number value.

Examples

- `random()` - Returns: 0.673411041443785
- `random(123)` - Returns: 0.06300625845328678
- `random(0)` - Returns: 0.06300625845328678
- `random(-122223)` - Returns: 0.3291484275937213

Note: You can call Random without ever starting a new random number sequence by specifying a negative seed parameter. If you do so, an internal seed will be generated using the system clock. The reason to start a new random number sequence by calling Random with a negative seed parameter, and then making subsequent calls to Random without a parameter (or with a positive parameter) is so that the report will look exactly the same every time it is previewed. In other words, it allows you to make use of random numbers, but to return a reproducible result.

Remainder(Integer a, Integer b)

This function is used to evaluate the remainder after numerator has been divided by a denominator.

Parameters

- a - An Integer value.
- b - An Integer value.

Return value

The return value is an Integer.

Examples

- `Remainder(7, 9)` - Returns 7.
- ```
  if(Remainder(@"CustomerID",2)==0) then
  return "0xffcc99"
  else
  return "0xccffff"
  ```
  This formula can distinguish odd numbers and even their backgrounds.

Round()

Round(Number a)

This function returns the integer portion after a specified number has been rounded.

Parameter

a - A Double value.

Return value

The return value is an Integer.

Example

The return value of the following statement is 3.00.

`Round(2.754)`

Round(Number a, integer b)

This function is used to round to a specified scale determined by argument b for Double value a.

Parameters

- a - A Double value.
- b - A Scale value.

Return value

The return value is a Number.

Example

The return value of the following statement is 2.46.

`Round(2.4576, 2)`

Round(Number a, integer b, integer c)

This function is used to round to a specified scale determined by argument b and a specified rounding mode determined by argument c for Double value a.

Parameters

- a - A double value which will be rounded.
- b - The rounded scale.
- c - The rounding mode.

  c is used to represent the following different seven rounding modes, which can be specified as an integer between 0 and 6.
  - ROUND_UP = 0
    Rounding mode to round away from zero. Always increments the digit prior to a non-zero discarded fraction. Note that this rounding mode never decreases the magnitude of the calculated value.
  - ROUND_DOWN = 1
    Rounding mode to round towards zero. Never increments the digit prior to a discarded fraction (that is, truncates). Note that this rounding mode never increases the magnitude of the calculated value.
  - ROUND_CEILING = 2
    Rounding mode to round towards positive infinity. If the value is positive, behaves as for ROUND_UP; if negative, behaves as for ROUND_DOWN. Note that this rounding mode never decreases the calculated value.
  - ROUND_FLOOR = 3
    Rounding mode to round towards negative infinity. If the BigDecimal is positive, behave as for ROUND_DOWN; if negative, behave as for ROUND_UP. Note that this rounding mode never increases the calculated value.
  - ROUND_HALF_UP = 4
    Rounding mode to round towards "nearest neighbor" unless both neighbors are equidistant, in which case round up. Behaves as for ROUND_UP if the discarded fraction is >= .5; otherwise, behaves as for ROUND_DOWN. Note that this is the rounding mode that most of us were taught in grade school.
  - ROUND_HALF_DOWN = 5
    Rounding mode to round towards "nearest neighbor" unless both neighbors are equidistant, in which case round down. Behaves as for ROUND_UP if the discarded fraction is > .5; otherwise, behaves as for ROUND_DOWN.
  - ROUND_HALF_EVEN = 6
    Rounding mode to round towards the "nearest neighbor" unless both neighbors are equidistant, in which case, round towards the even neighbor. Behaves as for ROUND_HALF_UP if the digit to the left of the discarded fraction is odd; behaves as for ROUND_HALF_DOWN if it's even. Note that this is the rounding mode that minimizes cumulative error when applied repeatedly over a sequence of calculations

Return Value

The return value is a Number.

Example

The return value of the following statement is 2.03.

`Round(2.021, 2, 0)`

Round (Number a, Integer b, String c)

This function is use to round to a specified scale determined by argument b using the rounding mode c.

Parameters

- a - A double value which will be rounded.
- b - The rounded scale.
- c - The rounding mode. It can be Up (0), DOWN (1), CEILING (2), FLOOR (3), HALF_UP (4), HALF_DOWN (5), and HALF_EVEN (6). See the previous section about the detailed description of each rounding mode.

Return Value

The return value is a number.

Example

The return value of the following statement is 123.457.

`Round(123.4567, 3, 'UP')`

Note: The digits of the decimal fraction of the rounded number must be bigger than the scale specified in the formula.

RunningAvg()

Function

- RunningAvg(field_variable);
- RunningAvg(field_variable, groupby);

Description

This function takes the average of all the numerical values in a field on running records. For details about the arguments, return type, and examples, refer to Average().

RunningCount()

Function

- RunningCount(field_variable);
- RunningCount(field_variable, groupby);

Description

This function takes a total count on running records of the values in a field. For details about the arguments, return type, and examples, refer to Count().

RunningDistinctCount()

Function

- RunningDistinctCount(field_variable);
- RunningDistinctCount(field_variable, groupby);

Description

This function takes a total count of all the distinct fields in a report on running records. For details about the arguments, return type, and examples, refer to DistinctCount().

RunningMaximum()

Function

- RunningMaximum(field_variable);
- RunningMaximum(field_variable, groupby);

Description

This function finds the largest numerical value in a field on running records. For details about the arguments, return type, and examples, refer to Maximum().

RunningMinimum()

Function

- RunningMinimum(field_variable);
- RunningMinimum(field_variable, groupby);

Description

This function finds the smallest numerical value in a field on running records. For details about the arguments, return type, and examples, refer to Minimum().

RunningSum()

Function

- RunningSum(field_variable);
- RunningSum(field_variable, groupby);

Description

This function sums up all the numerical values in a field on running records. For details about the arguments, return type, and examples, refer to Sum().

sgn(data)

Returns the sign value of the given number (whose value is data). If data is greater than 0, it will return 1. If data is less than 0, it will return -1. If data is equal to 0, it will return 0.

Parameter

data - The value you want to know the sign of.

Return value

1, 0 or -1.

Examples

- `sgn(2)` - Returns 1.
- `sgn(-23)` - Returns -1.
- `sgn(0)` - Returns 0.

sin(number)

Returns a number specifying the sine of an angle given in radians. It takes a right-angle triangle, and returns the length of the side opposite to the specified angle divided by the length of the hypotenuse.

Parameter

number - An angle in number of radians.

Return value

Number value between -1 and 1.

Examples

- `sin(1)` - Returns 0.84 (rounded to 2 decimals). This is the sine of 1 radian.
- `sin(60*pi/180)` - Returns 0.87 (rounded to 2 decimals). This is the sine of 60 degrees. Before taking the sine, the angle is converted to radians by multiplying by pi/180.

sqr(number)

Returns the square root of a given number. If the value of the number is less than 0, this function returns NaN. It is designed to work like the Java Math function of sqrt().

Parameter

number - A Number value greater than or equal to 0.

Return value

Number value.

Example

`sqr(100)` - Returns 10.

StdDev()

- StdDev(a[ ])
- StdDev(field)
- StdDev(field, groupby)

This function computes the standard deviation of the values referred to by the argument.

Parameters

- a - A Number array.
- field - Values of DBField or formula field.
- groupby - A constant string which indicates the DBField, formula or parameter grouped by.

Return value
  ```

Returns a Number value.

Examples

- `Number x = StdDev([1.0, 2.0, 3.0, 4.5, 45.0, 67.0])`
- `Number x[6] = [1.0, 2.0, 3.0, 4.5, 45.0, 67.0];`
  `Number y = StdDev( x)`
- `Number x = StdDev(@dbfield)`
- `Number x = StdDev(@formula)`
- `Number x = StdDev(@dbfield, @dbfield1)`
- `Number x = StdDev(@dbfield, @formula_groupby)`
- `Number x = StdDev(@dbfield, @parameter)`

Note: If a field is assigned to a variable, for example x, the variable x will lose the characteristic of representing a group of values. The following formula will then be treated as incorrect.

```
Number x = @dbfield;
Number y = StdDev( x); // system will prompt you that there is no such kind of function.
```

Sum()

- Sum(Number a[ ])
- Sum(Integer a[ ])
- Sum(Currency a[ ])
- Sum(DBField field)
- Sum(DBField, String groupby)

This function computes the sum of all the values referred to by the argument.

Parameters

- a - A Number, Integer or Currency array.
- field - Values of DBField or formula field.
- groupby - A constant string which indicates the DBField, formula or parameter grouped by.

Return value

The returned value type is dependent on the parameter. It can be Number, Integer or Currency.

Examples

- `Number x = Sum([1, 2, 3.0, 4.5, 45, 67])`
- `Number x[6] = [1, 2, 3.0, 4.5, 45, 67];`
  `Number y = Sum(x)`
- `Currency x = Sum([$1, $2, $3.0, $4.5, $45, $67])`
- `Currency x[] = [$1, $2, $3.0, $4.5, $45, $67];`
  `Currency y = Sum(x)`
- `Number x = Sum(@dbfield)`
- `Number x = Sum(@formula)`
- `Number x = Sum(@dbfield, @dbfield_groupby)`
- `Number x = Sum(@dbfield, @formula_groupby)`
- `Number x = Sum(@dbfield, @parameter)`
- `Currency x = Sum(@dbfield)`
- `Currency x = Sum(@formula)`
- `Currency x = Sum(@formula, @dbfield_groupby)`
- `Currency x = Sum(@formula, @formula_groupby)`
- `Currency x = Sum(@formula, @parameter)`

Note: If a field is assigned to a variable, for example x, the variable x will lose the characteristic of representing a group of values. So, the following formula will be treated as incorrect.

```
Number x = @dbfield;
Number y = Sum( x); // system will prompt you that there is no such kind of function.
```

tan(number)

Returns a number specifying the tangent of an angle given in radians. It takes a right-angle triangle, and returns the length of the side opposite the specified angle divided by the length of the side adjacent to the angle.

Parameter

number - An angle in the form of number of radians.

Return value

Number value.

Examples

- `tan(1)` - Returns 1.5574 (rounded to 4 decimals). This is the tangent of 1 radian.
- `tan(45 * pi / 180)` - Returns 1.0. This is the tangent of 45 degrees. Before taking the tangent, the angle is first converted to radians by multiplying it with pi/180.

ToBinary(Integer a)

This function changes a specified integer to its binary form.

Parameter

a - A BigInt value.

Return value

The return value is a String.

Example

The return value of the following statement is 1011.

```
ToBinary(11)
```

ToHex(Integer a)

This function returns the HEX form of a specified value.

Parameter

a - A BigInt value.

Return value

The return value is a String.

Example

The return value of the following statement is 20.

```
ToHex(32)
```

ToInt(number a)

This function changes a specified number to an Integer.

Parameter

a - A Number value.

Return value

The return value is an Integer.

Example

The return value of the following statement is 100.

```
ToInt(100.1)
```

ToOctal(Integer a)

This function returns the octal form of a specified value.

Parameter

a - A BigInt value.

Return value

The return value is a String.

Example

The return value of the following statement is 64.

```
ToOctal(52)
```

Truncate()

Truncate(Number a)

This function returns a number by truncating the decimal portion.

Parameter

a - A Double value to be truncated.

Return value

The return value is a Number.

Example

The return value of the following statement is 25.00.

```
Truncate(25.4321)
```

Truncate(Number a, Integer b)

This function returns a number by truncating the number at the decimal point. The number is truncated to the decimal place indicated by argument b, and the function returns a fractional number.

Parameters

- a - A Double value to be truncated.
- b - A Double value indicating the number of the decimal places.

Return value

The return value is a Number.

Example

The return value of the following statement is 25.430.

```
Truncate(25.432, 2)
```

Variance()

- Variance(Number a[ ])
- Variance(Currency a[ ])
- Variance(DBField field)
- Variance(DBField field, String groupby)

This function computes the variance of all the values referred to by the argument.

Parameters

- a - A Number array.
- field - Values of DBField or formula field.
- groupby - A constant string which indicates the DBField, formula or parameter grouped by.

Return value

Returns a Number type value.

Examples

- `Number x = Variance([1.0, 2.0, 3.0, 4.5, 45.0, 67.0])`
- `Number x[6] = [1.0, 2.0, 3.0, 4.5, 45.0, 67.0];`
  `Number y = Variance(x)`
- `Number x = Variance(@dbfield)`
- `Number x = Variance(@formula)`
- `Number x = Variance(@dbfield, @dbfield_groupby)`
- `Number x = Variance(@dbfield, @formula_groupby)`
- `Number x = Variance(@dbfield, @parameter)`

Note: If a field_variable is assigned to a variable, for example x, the variable x will lose the characteristic of representing a group of values. So, the following formula will be treated as incorrect.

```
Number x = @dbfield;
Number y = Variance(x); // system will prompt you that there is no such kind of function.
```

O-Q

String

Functions belonging to the String functions. Click the links for further information.

- [ord()](/en/function.ord.html)
- [parse_str()](/en/function.parse_str.html)
- [print()](/en/function.print.html)
- [printf()](/en/function.printf.html)
- [quotemeta()](/en/function.quotemeta.html)
- [sprintf()](/en/function.sprintf.html)
- [sscanf()](/en/function.sscanf.html)
- [str_contains()](/en/function.str_contains.html)
- [str_ends_with()](/en/function.str_ends_with.html)
- [str_getcsv()](/en/function.str_getcsv.html)
- [str_ireplace()](/en/function.str_ireplace.html)
- [str_pad()](/en/function.str_pad.html)
- [str_replace()](/en/function.str_replace.html)
- [str_rot13()](/en/function.str_rot13.html)
- [str_shuffle()](/en/function.str_shuffle.html)
- [str_split()](/en/function.str_split.html)
- [str_starts_with()](/en/function.str_starts_with.html)
- [str_word_count()](/en/function.str_word_count.html)
- [strcasecmp()](/en/function.strcasecmp.html)
- [strcmp()](/en/function.strcmp.html)
- [strcoll()](/en/function.strcoll.html)
- [strcspn()](/en/function.strcspn.html)
- [strip_tags()](/en/function.strip_tags.html)
- [stripcslashes()](/en/function.stripcslashes.html)
- [stripslashes()](/en/function.stripslashes.html)
- [stripos()](/en/function.stripos.html)
- [stristr()](/en/function.stristr.html)
- [strlen()](/en/function.strlen.html)
- [strpbrk()](/en/function.strpbrk.html)
- [strpos()](/en/function.strpos.html)
- [strrchr()](/en/function.strrchr.html)
- [strrev()](/en/function.strrev.html)
- [strripos()](/en/function.strripos.html)
- [strrpos()](/en/function.strrpos.html)
- [strspn()](/en/function.strspn.html)
- [strstr()](/en/function.strstr.html)
- [strtok()](/en/function.strtok.html)
- [strtolower()](/en/function.strtolower.html)
- [strtoupper()](/en/function.strtoupper.html)
- [strtr()](/en/function.strtr.html)
- [substr()](/en/function.substr.html)
- [substr_compare()](/en/function.substr_compare.html)
- [substr_count()](/en/function.substr_count.html)
- [substr_replace()](/en/function.substr_replace.html)
- [trim()](/en/function.trim.html)
- [ucfirst()](/en/function.ucfirst.html)
- [ucwords()](/en/function.ucwords.html)
- [vfprintf()](/en/function.vfprintf.html)
- [vprintf()](/en/function.vprintf.html)
- [vsprintf()](/en/function.vsprintf.html)
- [wordwrap()](/en/function.wordwrap.html)

AssertOne ⚠

The Assert Library for PHPUnit. Part of the Cake collection of the organization.

Dependencies
Assert, Core, Debug, Utility
Recent stable version
1.0.0 (2015-05-05)
Description
The Assert Library for PHPUnit.

Asynchronous

Asynchronous Task Library for and on a message or log. "asynchronous" [...]

- Version 0.1 (2012-08-14) Initial release. Only [...]
- Version 0.2 (2012-08-15) Fixed [...]

Dependencies
Base, Core, Log, Mail, Queue, Utility
Recent stable version
0.2.0 (2012-08-15)
Description
Asynchronous Task Library for and on a message or log.

Authorizer

Provides access control functionality. Authorizer uses the PHP-CRUD-API-Middleware and the Slim framework.

- Version 0.1.0 (2020-05-05) Initial release.

Dependencies
Slim, Middleware
Recent stable version
0.1.0 (2020-05-05)
Description
The Slim Access Control Authorizer.

AutoTest

The Cake Test Suite for PHPUnit. Part of the Cake collection of the organization.

Dependencies
Core, Console, Debug, Utility
Recent stable version
1.0.0 (2015-05-05)
Description
The Cake Test Suite for PHPUnit.

Auth/Yubikey

Yubikey authentication for PHP.

- Version 0.1.0 (2010-04-14) Initial release.
- Version 0.1.1 (2010-04-14) Fixed [...]
- Version 0.1.2 (2010-04-14) Fixed [...]

Dependencies
Base, Net_URL, Net_Socket, HTTP_Request
Recent stable version
0.1.2 (2010-04-14)
Description
Yubikey authentication for PHP.

AutoTwitter

Twitter Bot Library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Twitter Bot Library.

Auxiliary

Auxiliary Functions for Cake Framework.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Auxiliary Functions for Cake Framework.

Bcrypt/Crypt

Bcrypt/Crypt is a wrapper for the crypt() function in PHP.

- Version 0.1.0 (2011-12-12) Initial release.
- Version 0.1.1 (2011-12-12) Fixed [...]
- Version 0.1.2 (2011-12-12) Fixed [...]

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.2 (2011-12-12)
Description
Bcrypt/Crypt is a wrapper for the crypt() function in PHP.

Bcrypt/Hash

Bcrypt/Hash is a wrapper for the password_hash() function in PHP.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash is a wrapper for the password_hash() function in PHP.

Bcrypt/Compatibility

Bcrypt/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions.

Bcrypt/Compatibility/Legacy

Bcrypt/Compatibility/Legacy provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Crypt and Bcrypt/Hash libraries.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Compatibility/Legacy provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Crypt and Bcrypt/Hash libraries.

Bcrypt/Compatibility/Legacy/Crypt

Bcrypt/Compatibility/Legacy/Crypt provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Crypt library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Compatibility/Legacy/Crypt provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Crypt library.

Bcrypt/Compatibility/Legacy/Hash

Bcrypt/Compatibility/Legacy/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Compatibility/Legacy/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Compatibility/Legacy/Hash/Crypt

Bcrypt/Compatibility/Legacy/Hash/Crypt provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Crypt and Bcrypt/Hash libraries.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Compatibility/Legacy/Hash/Crypt provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Crypt and Bcrypt/Hash libraries.

Bcrypt/Compatibility/Legacy/Hash/Hash

Bcrypt/Compatibility/Legacy/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Compatibility/Legacy/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Compatibility

Bcrypt/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions.

Bcrypt/Hash/Compatibility/Legacy

Bcrypt/Hash/Compatibility/Legacy provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Compatibility/Legacy provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Crypt

Bcrypt/Hash/Crypt provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Crypt library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Crypt provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Crypt library.

Bcrypt/Hash/Crypt/Compatibility

Bcrypt/Hash/Crypt/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Crypt and Bcrypt/Hash libraries.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Crypt/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Crypt and Bcrypt/Hash libraries.

Bcrypt/Hash/Hash/Compatibility

Bcrypt/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Compatibility/Legacy

Bcrypt/Hash/Hash/Compatibility/Legacy provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Compatibility/Legacy provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash

Bcrypt/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Compatibility

Bcrypt/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash

Bcrypt/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Compatibility

Bcrypt/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash

Bcrypt/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Compatibility

Bcrypt/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log, Utility
Recent stable version
0.1.0 (2011-12-12)
Description
Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility

Bcrypt/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Hash/Compatibility provides compatibility functions for the password_hash() and the password_verify() functions in PHP 5.3.7 and later versions. This version supports the old hashes created by the Bcrypt/Hash library.

- Version 0.1.0 (2011-12-12) Initial release.

Dependencies
Base, Core, Log,

- a - A Currency value to be converted.
- b - A String indicating the format for displaying the value in a.
- c - The number of decimal places to be converted.

Return value

The return value is a String.

Example

The return value of the following statement is $123.5.

`ToText($123.456, "$##0.00", 1)`

ToText(Currency a, String b, Integer c, String d)

This function converts a Currency value to a String.

Parameters

- a - A Currency value to be converted.
- b - A String indicating the format for displaying the value in a.
- c - The number of decimal places to be converted.
- d - A single character string indicating the character to be used to separate thousands in a.

Return value

The return value is a String.

Example

The return value of the following statement is $1*234.57.

`ToText($1234.57, "$#,###.000", 2, "*")`

ToText(Currency a, String b, Integer c, String d, String e)

This function converts a Currency value to a String.

Parameters

- a - A Currency value to be converted.
- b - A String indicating the format for displaying the value in a.
- c - The number of decimal places to be converted.
- d - A single character string indicating the character to be used to separate thousands in a.
- e - A single character string indicating to be used as a decimal separator.

Return value

The return value is a String.

Examples

- The return value of the following statement is $4,567.123.
  `ToText($4567.123, "$#,##0.00", 3, "," , ".")`
- The return value of the following statement is $4&567*1.
  `ToText($4567.123, "$#,##0.00", 1, "&", "*")`

ToText(Currency a, Integer c)

This function converts a Currency value to a String.

Parameters

- a - A Currency value to be converted.
- c - The number of decimal places to be converted.

Return value

The return value is a String.

Example

The return value of the following statement is $123.46.

`ToText($123.4567, 2)`

ToText(Currency a, Integer c, String d)

This function converts a Currency value to a String.

Parameters

- a - A Currency value to be converted.
- c - The number of decimal places to be converted.
- d - A single character string indicating the character to be used to separate thousands in a.

Return value

The return value is a String.

Example

The return value of the following statement is $1,234.57.

`ToText($1234.567, 2, ",")`

ToText(Currency a, Integer c, String d,String e)

This function converts a Currency value to a String.

Parameters

- a - A currency value to be converted.
- c - The number of decimal places to be converted.
- d - A single character string indicating the character to be used to separate thousands in a.
- e - A single character string indicating to be used as a decimal separator.

Return value

The return value is a String.

Example

The return value of the following statement is $1,234.57.

`ToText($1234.567, 2, ",", ".")`

ToText(Date a)

This function converts a Date value to a String.

Parameter

a - A Data value.

Return value

The return value is a String.

Example

Suppose the date is July 15,1999, the return value of the following statement is 15-Jul-99.

`ToText(ToDate(1999,7,15))`

ToText(Date a, String b)

This function converts a Date value to a String.

Parameters

- a - A Date value.
- b - A String that defines how the Data value in a to be formatted.

Return value

The return value is a String.

Example

Suppose the date is July 15,1999, the return value of the following statement is 15-Jul-99.

`ToText(ToDate(1999,7,15), "dd-MMM-yy")`

ToText(DateTime a)

This function converts a DateTime value to a String.

Parameter

a - A TimeStamp value to be converted.

Return value

The return value is a String.

Example

Suppose the date is July 15,1999 and time is 7:42:51, the return value of the following statement is 15-Jul-99 7:42:51 AM.

`ToText(ToDateTime(1999,7,15,7,42,51))`

ToText(DateTime a, String b)

This function converts a DateTime value to a String.

Parameters

- a - A TimeStamp value to be converted.
- b - A String that defines how the DateTime value in a to be formatted.

Return value

The return value is a String.

Example

Suppose the date is July 15,1999 and time is 7:42:51, the return value of the following statement is 07-15-99 7:42:51.

`ToText(ToDateTime(1999,7,15,7,42,51), "MM-dd-yy h:mm:ss")`

ToText(DateTime a, String b, String c)

This function converts a DateTime value to a String.

Parameters

- a - A TimeStamp value to be converted.
- b - A String that defines how the DateTime value in a to be formatted.
- c - A String to be used as a label for A.M. (morning) hours.

Return value

The return value is a String.

Example

Suppose the date is July 15,1999 and time is 7:42:51, the return value of the following statement is 15-Jul-99 7:42:51 AM.

`ToText(ToDateTime(1999,7,15,7,42,51), "dd-MMM-yy h:mm:ss a", "AM")`

ToText(DateTime a, String b, String c, String d)

This function converts a DateTime value to a String.

Parameters

- a - A TimeStamp value to be converted.
- b - A String that defines how the DateTime value in a to be formatted.
- c - A String to be used as a label for A.M. (morning) hours.
- d - A String to be used as a label for P.M. (evening) hours.

Return value

The return value is a String.

Examples

- Suppose the date is Oct. 7,1999 and time is 7:42:51, the return value of the following statement is 07-Oct-99 7:42:51 AM.
  `ToText(ToDateTime(1999,10,7,7,42,51), "dd-MMM-yy h:mm:ss a", "AM", "PM")`
- Suppose the date is Oct. 7,1999 and time is 18:42:51, the return value of the following statement is 07-Oct-99 6:42:51 PM.
  `ToText(ToDateTime(1999,10,7,18,42,51), "dd-MMM-yy h:mm:ss a","AM", "PM")`

ToText(Integer a)

This function converts a BigInt value to a String.

Parameter

a - A BigInt value to be converted.

Return value

The return value is a String.

Example

The return value of the following statement is 456.

`ToText(456)`

ToText(Integer a, String b)

This function converts a BigInt value to a String.

Parameters

- a - A BigInt value to be converted.
- b - A String value indicating the format for displaying the value in argument a.

Return value

The return value is a String.

Example

The return value of the following statement is 456.

`ToText(456, "###")`

ToText (Integer a, String b, Integer c)

This function converts a BigInt value to a String.

Parameters

- a - A BigInt value to be converted.
- b - A String indicating the format for displaying the value in a.
- c - The number of decimal places to be converted.

Return value

The return value is a String.

Example

The return value of the following statement is 456.00.

`ToText(456, "###", 2)`

ToText(Integer a, String b, Integer c, String d)

This function converts a BigInt value to a String.

Parameters

- a - A BigInt value to be converted.
- b - A String indicating the format for displaying the value in a.
- c - The number of decimal places to be converted.
- d - A single character string indicating the character to be used to separate thousands in a.

Return value

The return value is a String.

Example

The return value of the following statement is 1,234.00.

`ToText(1234, "#,###", 2, ",", ".")`

ToText(Integer a, String b, Integer c, String d, String e)

This function converts a BigInt value to a String.

Parameters

- a - A BigInt value to be converted.
- b - A String indicating the format for displaying the value in a.
- c - The number of decimal places to be converted.
- d - A single character string indicating the character to be used to separate thousands in a.
- e - A single character string indicating to be used as a decimal separator.

Return value

The return value is a String.

Example

The return value of the following statement is 1,234.00.

`ToText(1234, "#,###", 2, ",", ".")`

ToText(Integer a, Integer c)

This function converts a BigInt value to a String.

Parameters

- a - A BigInt value to be converted.
- c - The number of decimal places to be converted.

Return value

The return value is a String.

Example

The return value of the following statement is 123.00.

`ToText(123, 2)`

ToText(Integer a, Integer c, String d)

This function converts a BigInt value to a String.

Parameters

- a - A BigInt value to be converted.
- c - The number of decimal places to be converted.
- d - A single character string indicating the character to be used to separate thousands in a.

Return value

The return value is a String.

Example

The return value of the following statement is 1,234.00.

`ToText(1234, 2, ",")`

ToText(Integer a, Integer c, String d, String e)

This function converts a BigInt value to a String.

Parameters

- a - A BigInt value to be converted.
- c - The number of decimal places to be converted.
- d - A single character string indicating the character to be used to separate thousands in a.
- e - A single character string indicating to be used as a decimal separator.

Return value

The return value is a String.

Example

The return value of the following statement is 12,345.00.

`ToText(12345, 2, ",", ".")`

ToText(Number a)

This function converts a Double value to a String.

Parameter

a - A Double value to be converted.

Return value

The return value is a String.

Example

The return value of the following statement is 123.456.

`ToText(123.456)`

ToText(Number a, String b)

This function converts a Double value to a String.

Parameters

- a - A Double value to be converted.
- b - A String indicating the format for displaying the value in a.

Return value

The return value is a String.

Example

The return value of the following statement is 1236.46.

`ToText(1236.456, "###0.00")`

ToText(Number a, String b, Integer c)

This function converts a Double value to a String.

Parameters

- a - A Double value to be converted.
- b - A String indicating the format for displaying the value in a.
- c - The number of decimal places to be converted.

Return value

The return value is a string.

Example

The return value of the following statement is 123.46.

`ToText(123.456, "###.00", 2)`

ToText(Number a, String b, Integer c, String d)

This function converts a Double value to a String.

Parameters

- a - A Double value to be converted.
- b - A String indicating the format for displaying the value in a.
- c - The number of decimal places to be converted.
- d - A single character string indicating the character to be used to separate thousands in a.

Return value

The return value is a String.

Example

The return value of the following statement is 1,234.57.

`ToText(1234.567, "#,###.000", 2, ",")`

ToText(Number a, String b, Integer c, String d,String e)

This function converts a Double value to a String.

Parameters

- a - A Double value to be converted.
- b - A String indicating the format for displaying the value in a.
- c - The number of decimal places to be converted.
- d - A single character string indicating the character to be used to separate thousands in a.
- e - A single character string indicating to be used as a decimal separator.

Return value

The return value is a String.

Example

The return value of the following statement is 1,234.57.

`ToText(1234.567, "#,###.000", 2, ",", ".")`

ToText(Number a, Integer c)

This function converts a Double value to a String.

Parameters

- a - A Double value to be converted.
- c - The number of decimal places to be converted.

Return value

The return value is a String.

Example

The return value of the following statement is 123.46.

`ToText(123.456, 2)`

ToText(Number a, Integer c, String d)

This function converts a Double value to a string.

Parameters

- a - A Double value to be converted.
- c - The number of decimal places to be converted.
- d - A single character string indicating the character to be used to separate thousands in a.

Return value

The return value is a String.

Example

The return value of the following statement is 1,234.6.

`ToText(1234.567, 1, ",")`

ToText(Number a, Integer c, String d, String e)

This function converts a Double value to a String.

Parameters

- a - A Double value to be converted.
- c - The number of decimal places to be converted.
- d - A single character string indicating the character to be used to separate thousands in a.
- e - A single character string indicating to be used as a decimal separator.

Return value

The return value is a String.

Example

The return value of the following statement is 1,234,57.

`ToText(1234.567, 2, "," , ",")`

ToText(Time a)

This function converts a Time value to a String.

Parameter

a - A Time value.

Return value

The return value is a String.

Example

Suppose the time is 12:40:30, the return value of the following statement is 12:40:30 PM.

`ToText(ToTime(12,40,30))`

ToText(Time a, String b)

This function converts a Time value to a String.

Parameters

- a - A Time value to be converted.
- b - A String that defines how the value in a to be formatted.

Return value

The return value is a String.

Example

Suppose the time is 08:48:58, the return value of the following statement is 8:48:58.

`ToText(ToTime(8,48,58), "h:mm:ss")`

ToText(Time a, String b, String c)

This function converts a Time value to a String.

Parameters

- a - A Time value to be converted.
- b - A String that defines how the value in a to be formatted.
- c - A String to be used as a label for A.M. (morning) hours.

Return value

The return value is a String.

Example

Suppose the time is 8:48:58 in the morning, the return value of the following statement is 8:48:58 AM.

`ToText(ToTime(8,48,58), "h:mm:ss a", "AM")`

ToText(Time a, String b,String c, String d)

This function converts a Time value to a String.

Parameters

- a - A Time value to be converted.
- b - A String that defines how the value in a to be formatted.
- c - A String to be used as a label for A.M. (morning) hours.
- d - A String to be used as a label for P.M. (evening) hours.

Return value

The return value is a String.

Examples

- Suppose the time is 08:48:58, the return value of the following statement is 8:48:58 AM.
  `ToText(ToTime(8,48,58), "h:mm:ss a", "AM", "PM")`
- Suppose the time is 21:49:59, the return value of the following statement is 9:49:59 PM.
  `ToText(ToTime(21,49,59), "h:mm:ss a", "AM", "PM")`

Notes:

- When converting a Double value, Currency value or a BigInt value to a String, special characters used as format in argument b are following:

Symbol	Meaning	Notes
0	a digit	
*	a digit, zero shows as a star	Can't mix 0, *, and _ in same format
_	a digit, zero shows as a space	Can't mix 0, *, and _ in same format
#	a digit, zero shows as absent	
.	placeholder for decimal separator	
,	placeholder for grouping delimiter	Shows the interval to be used
;	separates formats	positive and negative.
-	if there is no explicit negative sign, - is prefixed	"0.00" -> "0.00;-0.00"
%	divide by 100 and show as percentage	
X	any other characters can be used in the prefix or suffix	

- **Time format syntax:** The time format is specified by means of a String time pattern. In this pattern, all ASCII letters are reserved as pattern letters, which are defined as the following:

Symbol	Meaning	Presentation	Example
G	era designator	Text	AD
y	year	Number	1996
M	month in year	Text & Number	July & 07
d	day in month	Number	10
h	hour in am/pm (1~12)	Number	12
H	hour in day (0~23)	Number	0
m	minute in hour	Number	30
s	second in minute	Number	55
S	millisecond	Number	978
E	day in week	Text	Tuesday
D	day in year	Number	189
F	day of week in month	Number	2 (2nd Wed in July)
w	week in year	Number	27
W	week in month	Number	2
a	am/pm marker	Text	PM
k	hour in day (1~24)	Number	24
K	hour in am/pm (0~11)	Number	0
z	time zone	Text	Pacific Standard Time
'	escape for text		
''	single quote		

ToWords()

ToWords(Currency a)

This function converts a Number field value or the result of a numeric calculation to words.

Parameter

a - A Currency value to be converted into words.

Return value

The return value is a String.

Example

The return value of the following statement is three and 15/100.

`ToWords($3.15)`

ToWords(Currency a, Integer b)

This function converts a Number field value or the result of a numeric calculation to words.

Parameters

- a - A Currency value to be converted into words.
- b - A BigInt value indicating the number of decimal places to be converted. This argument is optional.

Return value

The return value is a String.

Example

The return value of the following statement is three and 5/10.

`ToWords($3.45, 1)`

ToWords(Integer a)

This function converts a Number field value or the result of a numeric calculation to words.

Parameter

a - A BigInt value to be converted into words.

Return value

The return value is a String.

Example

The return value of the following statement is twenty and xx/100.

`ToWords(20)`

ToWords(Integer a, Integer b)

This function converts a Number field value or the result of a numeric calculation to words.

Parameters

- a - A BigInt value to be converted into words.
- b - A BigInt value indicating the number of decimal places to be converted. This argument is optional.

Return value

The return value is a String.

Example

The return value of the following statement is three and x/10.

`ToWords(3, 1)`

ToWords(Number a)

This function converts a Number field value or the result of a numeric calculation to words.

Parameter

a - A fractional number to be converted into words.

Return value

The return value is a String.

Example

The return value of the following statement is ten and 12/100.

`ToWords(10.12)`

ToWords(Number a, Integer b)

This function converts a Number field value or the result of a numeric calculation to words.

Parameters

- a - A Number value to be converted into words.
- b - A BigInt value indicating the number of decimal places to be converted. This argument is optional.

Return value

The return value is a String.

Example

The return value of the following statement is three and 1/10.

`ToWords(3.14, 1)`

Translate()

Translate(String a)

This function searches for the NLS translation of current field value in the bound data mapping file.

- **Parameter**
  a - A String to which a corresponding data mapping file can be bound.
- **Return value**
  The return value is a String. It is the NLS translation of the field value if a data mapping file is bound to it and the NLS translation can be found, or null if the string is null, or the string itself when the corresponding NLS translation cannot be found in the data mapping file or the file cannot be found.
- **Example**
  The return value of the following statement is "••" and the locale is China if the language of the data mapping file bound to it is Chinese and it contains the mapping: ["USA", "••"].
- `Translate(@country)`, where @country has the value "USA".

Translate(String a, String b)

This function accesses the corresponding data mapping file dynamically according to the language selected from the language drop-down list on the NLS language toolbar and searches for the NLS translation of String b in the data mapping file.

- **Parameters**
  - a - A String, which is the prefix of the data mapping file name. For example, if the name is Category_de_DE.properties, set the string as Category (without the language and locale part).
  - b - A String indicating the string to be translated.
- **Return value**
  The return value is a String. It is the NLS translation of the field value if a data mapping file is bound to it and the NLS translation can be found, or null if the string is null, or the string itself when the corresponding NLS translation cannot be found in the data mapping file or the file cannot be found.
- **Examples**
  The return value of the following statement is "••" and the locale is China if the language of the data mapping file bound to it is Chinese and it contains the mapping: ["USA", "••"].
  - `Translate("mapping_file_country", "USA")`
  - `Translate("mapping_file_country", @country)`, where @country has the value "USA".

Trim(String a)

This function is used to remove the leading and the trailing spaces from string arguments.

Parameter

a - A String value.

Return value

The return value is a String.

Example

The return value of the following statement is he is a boy.

`Trim("he is a boy")`

TrimLeft(String a)

This function is used to remove the spaces on the left side of the string.

Parameter

a - A String value to be trimmed.

Return value

The return value is a String.

Example

The return value of the following statement is he is a boy.

`TrimLeft("he is a boy")`

TrimRight(String a)

This function is used to remove the spaces on the right side of the string.

Parameter

a - A String value to be trimmed.

Return value

The return value is a String.

Example

The return value of the following statement is he is a boy.

`TrimRight("he is a boy")`

UpperCase(String a)

This function converts all letters in string a to uppercase letters.

Parameter

a - A String value.

Return value

The return value is a String.

Example

The return value of the following statement is HE IS A BOY.

`UpperCase("he is a boy")`

Val(String a)

This function reads a string containing numbers (for example, address, phone or social security number), and converts them to a decimal value. Val stops reading when it finds the first character in the string.

Parameter

a - A String value.

Return value

The return value is a Number.

Example

The return value of the following statement is 63,550,513.00.

`Val("63550513Mr.Tom")`

Others

Besides the above mentioned function types, JReport has some other functions as listed below:

- Choose()
- currentBurstingSchema()
- eqv()
- getLanguage()
- getSecurityContext()
- isAll()
- isCountry()
- IsNoRecord()
- IsNull()
- isNumeric()
- isRunBursting()
- Next()
- openBinFile()
- openBinURL()
- openTxtFile()
- openTxtURL()
- Prev()
- reportName()
- Switch()
- toBool()
- toNumber()
- xor()

Choose(Integer, Array)

Returns a value from the array based on the value of Integer. For example, if integer is 0, it returns the first element in the array, and if index is 1 it returns the second element in the array.

Parameters

- Integer - A Number or numeric expression that specifies the index of the element. The minimum value is 0, and the maximum is the number of available elements minus 1. If it is out of bounds, no value will be returned.
- Array - An array that contains all the available elements to choose from. All elements must be of the same type. A choice can be any simple type (Number, Currency, String, Boolean, Date, Time or DateTime), or range type (Number Range, Currency Range, String Range, Date Range, Time Range or DateTime Range), but it may not be an array.

Return value

A value from the element in the given array. The type of the returned value is the same as the type of the element.

Examples

- `Choose(1,["Poor","Fair","Good","Excellent"])` - Returns Fair.
- `Choose(2,["1 to 10","11 to 20", "21 to 30"])` - Returns 21 to 30.
- `Choose(2,[1,2,3,4,5,6])` - Returns 3.
- `Choose(2,[Todate('1998/5/4'),ToDate('1999/5/5'),ToDate('1996/5/6')])` - Returns 5/6/96.

currentBurstingSchema()

Returns the names of the applied bursting schemas in the current report. When isRunBursting() returns false, the returned value of currentBurstingSchema() is NULL.

Note: Formulas referring to currentBurstingSchema() are not allowed to be used in queries or datasets.

Return value

The return value is a String.

Example

If the current report has been applied two bursting schemas: VP and Manager, the return value of the following statement is *VP,Manager*:

```
currentBurstingSchema()
```

eqv(booleanx, booleany)

Returns a Boolean value of booleanx eqv booleany,

- if x = true and y = true, returns the value of true.
- if x = true and y = false, returns the value of false.
- if x = false and y = true, returns the value of false.
- if x = false and y = false, returns the value of true.

Parameters

- booleanx - A Boolean value.
- booleany - A Boolean value.

Return value

Boolean value.

Examples

- `eqv(true, true)` - Returns true.
- `eqv(true, false)` - Returns false.

getLanguage()

Returns the language name of the locale that the current running task is based on.

Return value

The return value will either be the empty string or a lowercase ISO 639 code.

Example

When the locale is set to en_US, the return value of the following statement is *en*:

```
getLanguage()
```

getSecurityContext()

Returns a security context object, which provides the method get() to get the Security Context instance from JReport Server or JReport Designer.

Return value

The return value is a DbSecurityContext object.

Example

```
Import userClass from "UserFunction";
userClass.getData(getSecurityContext(), @country, …);
```

isAll(Array)

Checks whether the value of the specified parameter is "ALL".

Return value

The return value is true or false.

Example

When the parameter PEndDate supports multiple values and its value is set to "All", the following statement returns true.

```
isAll(@PEndDate)
```

isCountry(String)

Compares the input parameter with that of the country setting of JVM based on default locale.

Parameter

The input string must be an uppercase 2-letter ISO 3166 code.

Return value

The return value is true or false.

Examples

- If getLanguage() return zh, isCountry("CN") will return *true*.
- If getLanguage() return en, isCountry("CN") will return *false*.

IsNoRecord()

This function tells whether a report has returned a record or not. If the report has no value returned, the function will return True, otherwise False.

Note: A formula which calls this function cannot be applied to queries, and this formula only takes effect when laying out this report which contains it.

Example

When using this function, you should refer to a DBField to identify whether this formula is a record level formula so that it can be calculated when fetching the record. For example,

```
@"Customers_Customer ID";
if(IsNoRecord())
  return "There is no data"
else
  return "Total number of records="+@"Count_Customer Name5"
```

In this example, if no data returned, a tip "There is no data" will be displayed. If data is returned, the Total number of records will be displayed.

IsNull(DBfield a)

This function tells whether a specified value (especially the value of a DBField) is null or not. If the value is Null, the function will return True, otherwise False.

Parameter

a - A specified value especially a value of a DBField.

Return value

The return value is True or False.

Examples

- The return value of the following statement is false.

  ```
  IsNull(2.5)
  ```

- If you build a report about customer order, suppose one of the values of the field "Shipped" is Null, the return value of the following statement is true.

  ```
  IsNull(@shipped)
  ```

- ```
  Integer Total=0;
  if (!isnull(@grandTotal))
  Total= Total+@grandTotal;
  else
  Total=Total+0;
  ```

  This formula can summarize the Grand Total while ignoring the null value or no-record column.

isNumeric(inString)

Returns a Boolean value if the inString is a math number string.

Parameter

inString - A String type value.

Return value

Boolean value.

Examples

- `isNumeric("true")` - Returns false.
- `isNumeric("false")` - Returns false.
- `isNumeric("1234")` - Returns true.
- `isNumeric("122 322323")` - Returns false.

isRunBursting()

Returns true if the current report is running based on a bursting schema, else returns false.

Note: Formulas referring to isRunBursting() are not allowed to be used in queries or datasets.

Return value

The return value is true or false.

Examples

- If a bursting report is scheduled to run based on a bursting schema, the return value of the following statement is *true*:

  ```
  isRunBursting()
  ```

- If a bursting report is scheduled to run a normal result, the return value of the following statement is *false*:

  ```
  isRunBursting()
  ```

Next()

Next(DBfield a)

This function returns the next value of the current DBField.

Parameter

a - A DBField value.

Return value

The return value is a DBField value.

Example

Suppose you build a report about customer orders. If you use this function on "Ship Date" and insert it into the Detail Section, then when you run the report, you will see after each record, the next records will be displayed according to the following statement.

```
Next(@"Ship Date")
```

Next(DBfield a, BigInt b)

This function returns the next Nth record decided by the argument b.

Parameters

- a - A DBField value.
- b - A BigInt value.

Return value

The return value is a DBField value.

Example

Suppose you build a report about customer orders. If you use this function on "Ship Date" and set the argument b as 1, then when you run the report, you will see after each record, the next first record will be displayed according to the following statement.

```
Next(@"Ship Date", 1)
```

Note: Due to some implementation limitation, the argument b cannot be equal to or larger than 2.

openBinFile(string a)

This function is used to open an image file which is saved in your file system according to the path specified by the parameter.

Parameter

a - A String value which indicates the full path of the image file.

Return value

The return value is a binary image file.

Example

Suppose you have an image file photo1.gif which is saved in the following directory `c:\images`. The following example will open this image file.

```
openBinFile("c:\\images\\photo1.gif")
```

openBinURL(string a)

This function is used to open a binary image file according to the URL which is specified by the parameter.

Parameter

a - A String value which indicates the URL of the image file.

Return value

The return value is a binary image file.

Example

Suppose you have an image file in the following URL `http://www.jinfonet.com/../../asset/images/Pic1.gif`. The following example will open this image file Pic1.gif.

```
openBinURL("http://www.jinfonet.com/../../asset/images/Pic1.gif")
```

openTxtFile(string a)

This function is used to open a text file which is kept in your file system according to the path specified by the parameter.

Parameter

a - A String value which indicates the full path of the text file.

Return value

The return value is a long varchar.

Example

Suppose you have a text file report.int in the following directory `C:\JReport\Designer`. The following example will open this text file report.ini.

```
openTxtFile("C:\\JReport\\Designer\\report.ini")
```

openTxtURL(string a)

This function is used to open a text file according to the URL specified by the parameter.

Parameter

a - A String value which indicates the URL of the text file.

Return value

The return value is a long varchar.

Example

Suppose you have a text file at the following URL http://www.jinfonet.com/JReport/report.ini. The following example will open the text file report.ini.

```
openTxtURL("http://www.jinfonet.com/JReport/report.ini")
```

Prev()

Prev(DBfield a)

This function returns the previous value of the current DBField.

Parameter

a - A DBField value.

Return value

The return value is a DBField value.

Example

Suppose you build a report about customer orders. If you use this function on "Ship Date" and insert it into the Detail Section, after the report has been run, you will see the previous records displayed after each record, according to the following statement.

```
Prev(@"Ship Date")
```

Prev(DBfield a, BigInt b)

This function returns the previous Nth record decided by the argument b.

Parameters

- a - A DBField value.
- b - A BigInt value.

Return value

The return value is a DBField value.

Example

Suppose you build a report about customer orders. If you use this function on "Ship Date" and set the argument b as 4, then when you run the report, you will see before each record, the previous 4th record displayed according to the following statement.

```
Prev(@"Ship Date", 4)
```

Note: Due to some implementation limitation, the argument b cannot be equal to or less than -2.

reportName()

This function returns the current report name.

Return value

The return value is a String.

Example

If the current report name is Employee Information, the return value of the following statement is Employee Information.

```
reportName()
```

Switch(Boolean[], Array)

The elements in the two parameters corresponding with each other. This function evaluates the elements in the first parameter from left to right, and returns element associated with the first element to evaluate to True. For example,

```
Switch([1, 2, 3], [a, b, c])
```

if 1 is true, Switch returns a. If 2 is true, Switch returns b. If 3 is true, Switch returns c.

Parameters

- Boolean[] - A Boolean type array, which contains the elements that are to be evaluated.
- Array - An array, which contains all the available values that may be returned.

Return value

One of the elements of the parameter array. The type of the returned value is the same as the element in the array.

Example

Insert the following function into the Detail section of a report,

```
Switch([@"Customers_Customer ID"< 5, @"Customers_Customer ID" > 50,true],["small",
"large", "medium"])
```

- if Customer ID < 5 is true, returns small.
- if Customer ID >50 is true, returns large.
- if 5 < Customer ID < 50 (true), returns medium.

toBool(number or currency)

Returns True if the parameter is positive or negative but not 0, and returns False if the parameter is 0.

Parameter

Can be either a Number, Currency value, or expression.

Return value

Boolean value.

Example

`toBool(@Discount)` - Returns false if the discount is 0. Otherwise, returns true.

toNumber(Boolean)

Returns 1 if the parameter is True, and returns 0 if the parameter is False.

Parameter

True or false.

Return value

1 or 0.

Examples

- `toNumber(True)` - Returns 1.
- `toNumber(False)` - Returns 0.

xor(booleanx, booleany)

Returns a Boolean value of booleanx Exclusive OR booleany,

- if x = true and y = true, returns the value of false.
- if x = true and y = false, returns the value of true.
- if x = false and y = true, returns the value of true.
- if x = false and y = false, returns the value of false.

Parameters

- booleanx - A Boolean value.
- booleany - A Boolean value.

Return value

Boolean value.

Examples

- `xor(true, true)` - Returns false.
- `xor(true, false)` - Returns true.
- `xor((@"Customer ID"<=20),(Remainder(@"Customer ID", 2) == 0));`
```

Operators

The Operators panel in the Formula Editor lists all the operators JReport provides. They are powerful and commonly used. The operators can be divided into four types: Math, Comparison, Boolean, and Others.

Click a link from the following to get details:

- Math
- Comparison
- Boolean
- Others

Math

Following are Math operators. You can go into them to get detailed information.

- x + y
- x - y
- x * y
- x / y
- -x
- x % y
- x ^ y

x + y

date x + integer y

When a Data value is added to an Integer value, the return value will be a Data value because the Integer value will have changed to a Data value.

Example

If the date is Oct. 15, 1999, the return value of the following statement is 10-25-99.

```
ToDate (1999, 10, 15) + 10
```

datetime x + integer y

When a DateTime value is added to an Integer value, the return value will be a DateTime value because the Integer value will have changed to a DateTime value.

Example

If the datetime is Aug. 10, 1999 10:21:30, the return value of the following statement is 1999-08-10 10:21:40.

```
ToDateTime (1999, 8, 10, 10, 21, 30) + 10
```

numeric x + y

Adds x and y, and numeric adds to numeric. The returning value's precision will correspond to the data type with the higher precision except when a BigInt value is added to a real number. In this case, the return value will be a BigDecimal.

Note: Each data type has its own precision. For the Integer type data, BigInt has the highest precision while integer has the lowest precision. For the real number type data, Currency has the highest precision while Float has the lowest precision. So, if two numeric data of different precision are added together, the return value will be the data type with the higher precision. For example, if an Integer is added to a Double, the return value must be a Double value.

Example

The return value of the following expression is 81.71.

```
55 + 26.71
```

string x + boolean y

When a String is added to a Boolean value, the return value will be a String because the Boolean value can be changed to a String value, while the String value cannot be changed to a Boolean value.

Example

The return value of the following statement is It is false.

```
"It is" + IsNull (3>2)
```

string x + currency y

When a String is added to a Currency value, the return value will be a String because the Currency value can be changed to a String value, while the String value cannot be changed to a Currency value.

Example

The return value of the following statement is I spent9.56.

```
"I spent" + ToText($9.56)
```

string x + date y

When a String is added to a Date value, the return value will be a String because the Date value can be changed to a String value, while the String value cannot be changed to a Date value.

Example

The return value of the following statement is It is 1999-10-15.

```
"It is"+ToDate (1999, 10, 15)
```

string x + datetime y

When a String is added to a DateTime value, the return value will be a String because the DateTime value can be changed to a String value, while the String value cannot be changed to a Date value.

Example

The return value of the following statement is It is 1999-10-15 09:37:15.

```
"It is" + CurrentDateTime ()
```

string x + integer y

When a String is added to an Integer value, the return value will be a String because the Integer value can be changed to a String value, while the String value cannot be changed to an Integer value.

Example

The return value of the following statement is The result is9.

```
"The result is" + 9
```

string x + number y

When a String is added to a Number value, the return value will be a String because the Number value can be changed to a String value, while the String value cannot be changed to a Number value.

Example

The return value of the following statement is The result is9.56.

```
"The result is" + 9.56
```

string x + string y

The return value is a String.

Examples

- The return value of the following statement is It is your bike.

  ```
  "It is" + "your bike."
  ```

- ```
  String URL="http://localhost:8888/jinfonet/runReport.jsp?";
  String Reports = "&jrs.report_sheet$report2=true";
  String Catalog = "&jrs.catalog=/SampleReports/SampleReports.cat";
  String ReportSet = "&jrs.report=/SampleReports/SaleStat_detail.cls";
  String resultType = "&jrs.result_type=8";
  String para = "&jrs.param$P_Year=@INNER";
  String para1 = "&jrs.param$p_AccountManagerName=" + "Jeremy Miner";
  ```

  This formula sets an URL, which can run one demo report on the JReport Server.

string x + text y

When a String is added to a Text value, the return value will be a String because the Text value can be changed to a String value, while the String value cannot be changed to a Text value.

Example

The return value of the following statement is It is 07-Jul-99 7:12:21 AM.

```
"It is" + ToText (ToDateTime (1999, 7, 7, 7, 12, 21), "dd-MMM-yy h:mm:ss a", "AM")
```

string x + time y

When a String is added to a Time value, the return value will be a String because the Time value can be changed to a String value, while the String value cannot be changed to a Time value.

Example

The return value of the following statement is It is 10:10:10.

```
"It is" + ToTime (10, 10, 10)
```

text x + boolean y

When a text value is added to a Boolean value, the return value will be a text value because the Boolean value can be changed to a text value, while the text value cannot be changed to a Boolean value.

Example

The return value of the following statement is falsefalse.

```
ToText(3<2) + IsNull (2.5)
```

text x + currency y

When a text value is added to a Currency value, the return value will be a text value because the Currency value can be changed to a text value, while the text value cannot be changed to a Currency value. Note that with this option, when a Currency value is changed to a text value, the symbol $ is not added.

Example

The return value of the following statement is false10.56.

```
ToText(3<2) + ToText($10.56)
```

text x + date y

When a text value is added to a Data value, the return value will be a text value because the Data value can be changed to a text value, while the text value cannot be changed to a Data value.

Example

If the date is Aug.10, 1999 and the time is 10:21:30, the return value of the following statement is 10:21:301999-08-10.

```
ToText(ToTime(10,21,30), "hh:mm:ss") + ToDate(1999,8,10)
```

text x + datetime y

When a text value is added to a DateTime value, the return value will be a text value because the DateTime value can be changed to a text value while the text value cannot be changed to a DateTime value.

Example

If the date is Aug.10, 1999 and the time is 10:21:30, the return value of the following statement is false1999-08-10 10:21:30.

```
ToText(3<2) + ToDateTime(1999,8,10,10,21,30)
```

text x + integer y

When a text value is added to an Integer value, the return value will be a text value because the Integer value can be changed to a text value, while the text value cannot be changed to an Integer value.

Example

The return value of the following statement is false10.

```
ToText(3<2) + 10
```

text x + number y

When a text value is added to a Number value, the return value will be a text value because the Number value can be changed to a text value, while the text value cannot be changed to a Number value.

Example

The return value of the following statement is false10.56.

```
ToText(3<2) + 10.56
```

text x + string y

The return value is a String.

Example

The return value of the following statement is 25.60is a number.

```
ToText(25.6, 2)+"is a number."
```

text x + text y

The return value is a text value.

Example

The return value of the following statement is 25.6and6.52.

```
ToText(25.6, 1)+"and"+ToText(6.52, 2)
```

text x + time y

When a text value is added to a Data value, the return value will be a text value because the Data value can be changed to a text value, while the text value cannot be changed to a Data value.

Example

If the date is Aug.10, 1999 and the time is 10:21:30, the return value of the following statement is 99-08-1010:21:30.

```
ToText(ToDate(1999,8,10), "yy-MM-dd") + ToTime(10,21,30)
```

time x + integer y

When a Time value is added to an Integer value, the return value will be a Time value because the Integer value will have changed to a Time value.

Example

If the time is 10:15:25, the return value of the following statement is 10:15:35.

```
ToTime (10, 15, 25) + 10
```

x - y

date x - date y

When a Data value subtracts a Data value, the return value will be an Integer. The result is the number of days between the two dates.

Examples

- If one date is 1999-12-12, and the other date is 1995-12-12, the return value of the following statement is 1461.

  ```
  ToDate(1999, 12, 12) - ToDate(1995, 12, 12)
  ```

- ```
  if ((todate(@"Due Date")-CurrentDate())<0)
  {return '0xff0000'}//red
  else
  {return '0x000000'}//black
  ```

  This formula sets values' background, in which way it can highlight report values that are out of range.

date x - integer y

When a Data value subtracts an Integer value, the return value will be a Data value because the Integer value will have changed into a Data value.

Example

If the date is Oct. 18, 1999, the return value of the following statement is 10/16/99.

```
ToDate(1999, 10, 18) - 2
```

datetime x - datetime y

When a DateTime value subtracts a DateTime value, the return value will be an Integer. The result is the number of seconds between the two DateTimes.

Example

If one timestamp is 1999-12-12 10:10:10, and the other timestamp is 1995-12-12 10:10:10, the return value of the following statement is 126230400.

```
ToDateTime(1999, 12, 12, 10, 10, 10) - ToDateTime(1995, 12, 12, 10, 10, 10)
```

datetime x - integer y

When a DateTime value subtracts an Integer value, the return value will be a DateTime value because the Integer value will have changed into a DateTime value.

Example

If the date is Oct. 18, 1999, the time is 10:10:10, the return value of the following statement is 1999-10-18 10:10:08.

```
ToDateTime(1999, 10, 18, 10, 10, 10) - 2
```

numeric x - numeric y

Subtracts y from x, and numeric subtracts numeric. The precision of the return value will correspond to the data type with the higher precision.

Note: Each data type has its own precision. For the Integer type data, BigInt has the highest precision while integer has the lowest precision. For the real number type data, Currency has the highest precision while float has the lowest precision. So, if two numeric datum of different precision subtracts each other, the return value will be of the data type with the higher precision. For example, if an Integer subtracts a Double, the return value will be a Double value.

Example

The return value of the following expression is 61.71.

```
87.71 - 26
```

time x - integer y

When a Time value subtracts an Integer value, the return value will be a Time value because the Integer value will have changed into a Time value.

Example

If the time is 10:10:10, the return value of the following statement is 10:10:08.

```
ToTime(10, 10, 10) - 2
```

time x - time y

When a Time value subtracts a Time value, the return value will be an Integer. The result is the number of seconds between the two times.

Example

If one time is 08:08:08, and the other time is 10:10:10, the return value of the following statement is 7322.

```
ToTime(10, 10, 10) - ToTime(8, 8, 8)
```

x * y

currency x * currency y

Multiplies x by y. When a Currency value is multiplied by a Currency value, the return value will be a Currency value.

Example

The return value of the following statement is 2.88.

```
$2.4 * $1.2
```

currency x * integer y

Multiplies x by y. When a Currency value is multiplied by an Integer value, the return value will be a Currency value.

Example

The return value of the following statement is 106.75.

```
$4.27 * 25
```

currency x * number y

Multiplies x by y. When a Currency value is multiplied by a Number value, the return value will be a Currency value.

Example

The return value of the following statement is 3.25.

```
$1.27 * 2.56
```

integer x * currency y

Multiplies x by y. When an Integer value is multiplied by a Currency value, the return value will be a Currency value.

Example

The return value of the following statement is 106.75.

```
25 * $4.27
```

integer x * integer y

Multiplies x by y. When an Integer value is multiplied by an Integer value, the return value will be an Integer value.

Example

The return value of the following statement is 15.

```
3 * 5
```

integer x * number y

Multiplies x by y. When an Integer value is multiplied by a Number value, the return value will be a Number value.

Example

The return value of the following statement is 104.25.

```
25 * 4.17
```

number x * currency y

Multiplies x by y. When a Number value is multiplied by a Currency value, the return value will be a Currency value.

Example

The return value of the following statement is 3.25.

```
2.56 * $1.27
```

number x * integer y

Multiplies x by y. When a Number value is multiplied by an Integer value, the return value will be a Number value.

Example

The return value of the following statement is 104.25.

```
4.17 * 25
```

number x * number y

Multiplies x by y. When a Number value is multiplied by a Number value, the return value will be a Number value.

Example

The return value of the following statement is 2.88.

```
2.4 * 1.2
```

x / y

currency x / currency y

Divides x by y. When a Currency is divided by a Currency value, the return value will be an Integer or a Currency value.

Example

The return value of the following statement is 1.77.

```
$4.52 / $2.56
```

currency x / integer y

Divides x by y. When a Currency is divided by an Integer, the return value will be a Currency value.

Example

The return value of the following statement is 1.28.

```
$2.56 / 2
```

currency x / number y

Divides x by y. When a Currency is divided by a Number value, the return value will be an Integer or a Currency value.

Example

The return value of the following statement is 1.79.

```
$4.52 / 2.52
```

integer x / currency y

Divides x by y. When an Integer is divided by a Currency value, the return value will be a Currency or an Integer.

Example

The return value of the following statement is 21.87.

```
56 / $2.56
```

integer x / integer y

Divides x by y. When an Integer is divided by an Integer, the return value will be an Integer or a Number value.

Example

`56 / 3` - Return 18.67

`10 / 3` - Return 3.33

integer x / number y

Divides x by y. When an Integer is divided by a Number value, the return value will be a Number or an Integer.

Example

The return value of the following statement is 21.88.

```
56 / 2.56
```

number x / currency y

Divides x by y. When a Number is divided by a Currency value, the return value will be a Currency value.

Example

The return value of the following statement is 1.77.

```
4.52 / $2.56
```

number x / integer y

Divides x by y. When a Number is divided by an Integer, the return value will be a Number value.

Example

The return value of the following statement is 0.51.

```
2.55 / 5
```

number x / number y

Divides x by y. When a Number is divided by a Number value, the return value will be an Integer or a Number value.

Example

The return value of the following statement is 1.77.

```
4.52 / 2.56
```

number x \ number y

Divides x by y, and returns an Integer result.

Examples

- `39.4 \ 3` - Return 13.00
- `10 \ 3` - Return 3.00

-x

currency -x

Returns the opposite value of x. If a Currency value is positive, the return value will be negative. If the Currency value is negative, the return value will be positive.

Examples

- The return value of the following statement is -$2.57.

  ```
  -$2.57
  ```

- The return value of the following statement is $2.57.

  ```
  -(-$2.57)
  ```

integer -x

Returns the opposite value of x. If an Integer value is positive, the return value will be negative. If the Integer value is negative, the return value will be positive.

Examples

- The return value of the following statement is -25.

  ```
  -25
  ```

- The return value of the following statement is 25.

  ```
  -(-25)
  ```

number -x

Returns the opposite value of x. If a Number value is positive, the return value will be negative. If the Number value is negative, the return value will be positive.

Examples

- The return value of the following statement is -5.28.

  ```
  -5.28
  ```

- The return value of the following statement is 5.28.

  ```
  -(-5.28)
  ```

x % y

currency x % currency y

Computes the percentage of dividing x by y. When a Currency value is divided by a Currency value, the return value will be an Integer or a Currency.

Example

The return value of the following statement is 176.56.

```
4.52%2.56
```

currency x % integer y

Computes the percentage of dividing x by y. When a Currency value is divided by an Integer value, the return value will be an Integer or a Currency.

Example

The return value of the following statement is 226.

```
$4.52%2
```

currency x % number y

Computes the percentage of dividing x by y. When a Currency value is divided by a Number value, the return value will be an Integer or a Currency.

Example

The return value of the following statement is 120.

```
$1.44%1.2
```

integer x % currency y

Computes the percentage of dividing x by y. When an Integer value is divided by a Currency, the return value will be an Integer or a Currency.

Example

The return value of the following statement is 40.

```
1%$2.5
```

integer x % integer y

Computes the percentage of dividing x by y. When an Integer value is divided by an Integer, the return value will be an Integer or a Number.

Example

The return value of the following statement is 77.78.

```
7%9
```

integer x % number y

Computes the percentage of dividing x by y. When an Integer value is divided by a Number, the return value will be a Number or an Integer.

Example

The return value of the following statement is 400.

```
6%1.5
```

number x % currency y

Computes the percentage of dividing x by y. When a Number value is divided by a Currency, the return value will be an Integer or a Currency value.

Example

The return value of the following statement is 500.

```
2.5%$0.5
```

number x % integer y

Computes the percentage of dividing x by y. When a Number value is divided by an Integer, the return value will be an Integer or a Number.

Example

The return value of the following statement is 50.

```
2.5%5
```

number x % number y

Computes the percentage of dividing x by y. When a Number value is divided by a Number, the return value will be an Integer or a Number.

Example

The return value of the following statement is 176.56.

```
4.52%2.56
```

x ^ y

number x ^ number y

Raise x to the power of y. y can be fractional, positive or negative. x can be fractional, but negative only when y is a whole number.

Examples

- The return value of the following statement is 4.

  ```
  2^2
  ```

- The return value of the following statement is 1626.9438030745541.

  ```
  7^3.8
  ```

Comparison

Following are Comparison operators. You can go into them to get detailed information.

- x > y
- x < y
- x == y
- x <> y or x != y

x > y

date x > date y

If x is greater than y, the return value will be true. If x is < and == y, the return value will be false.

Example

The return value of the following statement is true.

```
ToDate(1999, 10, 10) > ToDate(1999, 2, 5)
```

datetime x > datetime y

If x is greater than y, the return value will be true. If x is < and == y, the return value will be false.

Example

The return value of the following statement is true.

```
ToDateTime(1999, 10, 10, 10, 10, 10) > ToDateTime(1999, 5, 10, 8, 5, 5)
```

numeric x > numeric y

If x is greater than y, the return value will be true. If x is < and == y, the return value will be false.

Examples

- The return value of the following statement is true.
  ```
  3>2.56
  ```
- The return value of the following statement is false.
  ```
  5.29>7.55
  ```

string x > string y

If x is greater than y, the return value will be true. If x is < and == y, the return value will be false.

Examples

- The return value of the following statement is true.
  ```
  "string abcd" > "string abc"
  ```
- The return value of the following statement is false.
  ```
  "string abc" > "string abcd"
  ```

time x > time y

If x is greater than y, the return value will be true. If x is < and == y, the return value will be false.

Example

The return value of the following statement is true.

```
ToTime(10, 10, 10) > ToTime(8, 10, 10)
```

x < y

date x < date y

If x is less than y, the return value will be true. If x is > and == y, the return value will be false.

Example

The return value of the following statement is false.

```
ToDate(1999, 10, 10) < ToDate(1999, 2, 5)
```

datetime < datetime y

If x is less than y, the return value will be true. If x is > and == y, the return value will be false.

Example

The return value of the following statement is false.

```
ToDateTime(1999, 10, 10, 10, 10, 10) < ToDateTime(1999, 5, 10, 8, 5, 5)
```

numeric x < numeric y

If x is less than y, the return value will be true. If x is > and == y, the return value will be false.

Examples

- The return value of the following statement is true.
  ```
  2.56<3
  ```
- The return value of the following statement is false.
  ```
  7.55<5.88
  ```

string x < string y

If x is less than y, the return value will be true. If x is > and == y, the return value will be false.

Examples

- The return value of the following statement is true.
  ```
  "string abc" < "string abcd"
  ```
- The return value of the following statement is false.
  ```
  "string abcd" < "string abc"
  ```

time x < time y

If x is less than y, the return value will be true. If x is > and == y, the return value will be false.

Example

The return value of the following statement is false.

```
ToTime(10, 10, 10) < ToTime(8, 10, 10)
```

x == y

date x == date y

If x is equal to y, the return value is true. If x is not equal to y, the return value will be false.

Example

The return value of the following statement is false.

```
ToDate(1999, 10, 10) == ToDate(1999, 5, 10)
```

datetime == datetime y

If x is equal to y, the return value will be true. If x is not equal to y, the return value will be false.

Example

The return value of the following statement is false.

```
ToDateTime(1999, 10, 10, 5, 10, 10) == ToDateTime(1999,10, 10, 10, 10, 10)
```

numeric x == numeric y

If x is equal to y, the return value will be true. If x is not equal to y, the return value will be false.

Example

The return value of the following statement is false.

```
5==4
```

string x == string y

If x is equal to y, the return value is true. If x is not equal to y, the return value will be false.

Example

The return value of the following statement is false.

```
"string abcd" == "string abc"
```

time x == time y

If x is equal to y, the return value will be true. If x is not equal to y, the return value will be false.

Example

The return value of the following statement is false.

```
ToTime(9, 10, 10) == ToTime(10, 10, 10)
```

x <> y or x != y

date x <> date y or date x != date y

If x is not equal to y, the return value will be true. If x is equal to y, the return value will be false.

Example

The return value of the following statement is true.

```
ToDate(1999, 10, 10) != ToDate(1999, 5, 10)
```

datetime x <> datetime y or datetime x != datetime y

If x is not equal to y, the return value will be true. If x is equal to y, the return value will be false.

Example

The return value of the following statement is true.

```
ToDateTime(1999, 10, 10, 5, 10, 10) != ToDateTime(1999,10, 10, 10, 10, 10)
```

numeric x <> numeric y or numeric x != numeric y

If x is not equal to y, the return value will be true. If x is equal to y, the return value will be false.

Example

The return value of the following statement is false.

```
4 <> 4
```

string x <> string y or string x != string y

If x is not equal to y, the return value will be true. If x is equal to y, the return value will be false.

Example

The return value of the following statement is true.

```
"string abc" != "string abcd"
```

time x <> time y or time x != time y

If x is not equal to y, the return value will be true. If x is equal to y, the return value will be false.

Example

The return value of the following statement is true.

```
ToTime(9, 10, 10) <> ToTime(10, 10, 10)
```

Boolean

- x || y
- x && y
- !x

Below is a set of truth tables showing the logic operations for 3-valued logic.

A AND B	True	False	Null
True	True	False	Null
False	False	False	False
Null	Null	False	Null

A OR *B*	True	False	Null
True	True	True	True
False	True	False	Null
Null	True	Null	Null

A	NOT *A*
True	False
False	True
Null	Null

Boolean x || Boolean y

Returns true if x is true or y is true.

Examples

- The return value of the following statement is abc.

```
Integer s=4,d=8;
Integer f=6;
if ( f>s || f>d )
{
return "abc";
}
```

-
```
if ((@"Annual Sales" > 50000) ||(@"Annual Sales"<0)) then
"Customer needs attention"
else
"Normal customer"
```

-
```
If( ! isNull(@customerId) && ! isNull(@country)){
                If(@customerId > 10 || @country == 'USA'){
                                Integer newId =  @customerId + 1;
}
}
Else{
                // Some error processing logic here
}
```

Boolean x && Boolean y

Returns true if x is true and y is true.

Examples

- The return value of the following statement is abc.

```
Integer s=4,d=8;
Integer f=6;
if ( f>s && f<d )
{
return "abc";
}
```

-
```
if ((@"Annual Sales" <3000) && (@Customers_Country=="USA")) then
"Customer is in the USA and Annual Sales is less than 3000"
else
"Others"
```

! Boolean x

Returns true if x is false.

Example

The return value of the following statement is def.

```
Integer s=4, d=8;
Integer f=6;
if ( ! ( f>s ) )
return  "abc"
else
return "def"
```

Others

- x in y
- [x, y, z]
- x[i]
- x[i to j]
- x[i _to j]
- x[i to_ j]
- x[i _to_ j]
- x[upfrom j]
- x[upfrom_ j]
- x[upto j]
- x[upto_ j]
- if(b) then{...} else{...}
- return x

x in y

A statement used to tell whether x is in y, the return value is a boolean value.

Examples

- The return value of the following statement is false.

```
integer x[3] = [0,1,2];
integer xx = 3;
xx in x
```

-
```
integer range intary=[30 to 50];
if (@"Product_ID" in intary) then
return "Within the Range"
else
return "Out of Range"
```

-
```
if (@Customers_Region in ["CO","MT","UT","WY"]) then
"Rocky Mountain Region"
else
"Rest of Country"
```

[x, y, z]

A statement used to define an array.

Example

The return value of the following statement is 1996-11-27.

```
date d[3] = [toDate(1997, 11, 27), toDate(1996, 11, 27), toDate(1995, 11, 27)];
date dd[3];
dd = d;
dd[1]
```

x[i]

A statement used to index a certain value in an array.

Example

The return value of the following statement is 14:11:27.

```
time d[3] = [totime(12, 11, 27), totime(13, 11, 27), totime(14, 11, 27)];
time dd[3];
dd = d;
dd[2]
```

x[i to j]

A statement used to define an array.

Example

The return value of the following statement is true.

```
string range strary = ["a" to "z"];
string x = "d";
x in strary
```

x[i _to j]

It is used to specify a range of values greater than but not including the value i, and less than or equal to the value j. Both i and j are the type of Number.

Example

The return value of the following statement is true.

```
integer a[] = [1,1,2,3,4];
3 in a[2 _to 4];
```

x[i to_ j]

It is used to specify a range of values greater than or equal to the value i, and less than but not including the j value. Both i and j are the type of Number.

Example

The return value of the following statement is true.

```
integer a[] = [1,1,2,3,4];
3 in a[2 to_ 4];
```

x[i _to_ j]

It is used to specify a range of values greater than but not including the value i, and less than but not including the value j. Both i and j are the type of Number.

Example

The return value of the following statement is true.

```
integer a[] = [1,1,2,3,4];
3 in a[2 _to_ 4];
```

x[upfrom j]

It is used to specify a range of values greater than or equal to the value j. j is the type of Number.

Example

The return value of the following statement is false.

```
integer a[] = [1,1,2,3,4];
3 in a[upfrom 4]
```

x[upfrom_ j]

It is used to specify a range of values greater than but not including the value j. j is the type of Number.

Example

The return value of the following statement is true.

```
integer a[] = [1,1,2,3,4];
3 in a[upfrom_ 2];
```

x[upto j]

It is used to specify a range of values less than or equal to the value j. j is the type of Number.

Example

The return value of the following statement is true.

```
integer a[] = [1,1,2,3,4];
3 in a[upto 4];
```

x[upto_ j]

It is used to specify a range of values less than but not including the value j. j is the type of Number.

Example

The return value of the following statement is false.

```
integer a[] = [1,1,2,3,4];
4 in a[upto_ 4];
```

if(b) then{...} else{...}

Conditional statement.

Example

The return value of the following statement is abc.

```
integer s=5;
integer f=8;
if ( f>s )
then return "abc"
else return "def"
```

return x

Returns the value of X.

Example

The result of the following statement is 06/19/01.

```
return Today ()
```

Web Report Object Property Reference

This chapter is an overall introduction to the properties of the web report objects in Web Report Studio. Keep in mind the following while browsing each object for its properties:

- You will start off from a table listing all the properties for an object and what can be achieved with each property.
- Some properties are just used to show related information and cannot be edited.
- Some properties can be controlled by a formula or expression. If a formula is selected to control a property, it will be automatically generated as an expression prefixed by an @ sign. Also, some properties must be known by JReport Engine before a page break, so that it can lay out the object. In a formula or expression, the special field PageNumber is the switch for setting the page break calculation time. If PageNumber is used in the formula or expression, then the calculation time will be after the page break, if PageNumber is not used, the calculation time will be before the page break. For this reason, JReport will set the calculation time for controlling whether the formula or expression will be marked as before page break or after page break. For most properties, the property can be controlled by formula or expression at any time without concern about the page break.
- For properties with a hyperlink, you can find further explanation. The same property for different objects usually means the same thing, unless otherwise stated.

The following are object properties for web report included in the Inspector panel (displayed when a web report is running in the Edit Mode of Web Report Studio), which are listed in alphabetical order:

- Chart
- Crosstab
- DBField
- Filter control
- Formula field
- Image
- Label
- Multimedia object (OLE object)
- Navigation control
- Page footer panel
- Page header panel
- Parameter control
- Parameter field
- Parameer form control
- Report body
- Special field
- Table
- Tabular
- Web report

Chart

A chart mainly consists of three parts: the platform, paper, and legend. The chart platform is a container for other chart elements, and the chart paper holds walls, axes, data markers, axis values, and category names.

The following topics are detailed introduction to properties of the elements:

- Chart label
- Chart legend
- Chart paper
- Chart platform
- A detailed chart property reference

Chart label

The properties of a chart label are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. Enter a numeric value to change the position in inches. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. Enter a numeric value to change the position in inches. Data type: Float
Label	
Bottom Margin	Specifies the distance between the label text and the bottom border of the label, in inches. The four properties, Bottom Margin, Left Margin, Right Margin and Top Margin work together with the property Text Alignment. Enter a numeric value to change the margin. Data type: Float
Color	Specifies the color of the label, applied when the property Fill Type is set to color. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Fill Transparency	Specifies the transparency of the background, in percent. Enter a numeric value to change the transparency. Data type: Integer
Fill Type	Specifies the fill pattern for the label. Choose an option from the drop-down list. Data type: Integer
Font Face	Specifies the font face of the label text. Choose an option from the drop-down list. Data type: Enumeration
Font Rotation	Specifies the rotation angle of the label text around its center, in degrees. The default value is 0. Enter a numeric value to change the rotation. Data type: Float
Font Size	Specifies the font size of the label text. Enter an integer value to change the size. Data type: Integer
Font Style	Specifies the font style for the label text. Choose an option from the drop-down list. Data type: Enumeration
Left Margin	Specifies the distance between the label text and the left border of the label, in inches. The four properties, Bottom Margin, Left Margin, Right Margin and Top Margin work together with the property Text Alignment. Enter a numeric value to change the margin. Data type: Float
Right Margin	Specifies the distance between the label text and the right border of the label, in inches. The four properties, Bottom Margin, Left Margin, Right Margin and Top Margin work together with the property Text Alignment. Enter a numeric value to change the margin. Data type: Float
Text	Specifies the text of the label. Enter a string to change the text. Data type: String
Text Alignment	Specifies the alignment style of the label text. It works together with the four properties, Bottom Margin, Left Margin, Right Margin and Top Margin. Choose an option from the drop-down list. Data type: Enumeration
Texture Background Color	Specifies the background color of the texture, applied when the property Fill Type is set to texture. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Top Margin	Specifies the distance between the label text and the top border of the label, in inches. The four properties, Bottom Margin, Left Margin, Right Margin and Top Margin work together with the property Text Alignment. Enter a numeric value to change the margin. Data type: Float
Background	
Border Color	Specifies the color of the label border. Choose a color from the drop-down list, enter a hexadecimal RGB value (0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border End Caps	Specifies the ending style of the label border line. Choose an option from the drop-down list. Data type: Enumeration
Border Style	Specifies the line style of the label border. Choose an option from the drop-down list. Data type: Enumeration
Border Thickness	Specifies the width of the label border, in inches. Enter a numeric value to change the thickness. Data type: Float
Border Transparency	Specifies the transparency of the label border, in percent. Enter a numeric value to change the transparency. Data type: Integer
Border Type	Specifies the border type of the label. Choose an option from the drop-down list. Data type: Enumeration
Border Variable Dash	Specifies whether to resize the dash automatically. • **true** - If selected, the dash size will be adjusted automatically. • **false** - If selected, the dash size will be of fixed size. Data type: Boolean
Color	Specifies the color of the label background, applied when the property Fill Type is set to color. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Fill Transparency	Specifies the transparency of the label background, in percent. Enter a numeric value to change the transparency. Data type: Integer
Fill Type	Specifies the fill pattern for the label background. Choose an option from the drop-down list. Data type: Enumeration
Inset Bottom	Specifies the bottom position of the background area, measured in a percentage of the label height, from the bottom edge of the label. Data type: Float
Inset Left	Specifies the left position of the background area, measured in a percentage of the label width, from the left edge of the label. Data type: Float
Inset Right	Specifies the right position of the background area, measured in a percentage of the label width, from the right edge of the label. Data type: Float
Inset Top	Specifies the top position of the background area, measured in a percentage of the label height, from the top edge of the label. Data type: Float
Icon	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (0xCCAA00), or select Custom from the drop-down list to specify the color. Data type: String
Border End Caps	Specifies the ending style of the icon border line. Choose an option from the drop-down list. Data type: Enumeration
Border Joint	Specifies the joint style of the icon border line. Choose an option from the drop-down list. Data type: Enumeration
Border Outlined	Specifies whether to show the icon border line in Outline Path. If the property is set to true, the icon border line will be shown in Outline Path; otherwise, in Fill Path. Data type: Boolean
Border Style	Specifies the line style of the icon borders. Choose an option from the drop-down list. Data type: Enumeration
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Border Transparency	Specifies the transparency of the icon borders, in percent. Enter a numeric value to change the transparency. Data type: Integer
Border Variable Dash	Specifies whether to resize the dash automatically. If the property is set to true, the dash size will be adjusted automatically; otherwise, the dash size will be of fixed size. Data type: Boolean
Color	Specifies the color of the icon background, applied when the property Fill Type is set to color. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to specify the color. Data type: String
Fill Transparency	Specifies the transparency of the icon background, in percent. Enter a numeric value to change the transparency. Data type: Integer
Fill Type	Specifies the fill pattern for the icon. Choose an option from the drop-down list. Data type: Enumeration
Gradient End Color	Specifies the color of the point where the gradient ends, applied when the property Fill Type is set to gradient. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to specify the color. Data type: String
Gradient End X	Specifies the horizontal position where the gradient ends, measured in a percentage of the icon width. Applied when the property Fill Type is set to gradient. Data type: Integer
Gradient End Y	Specifies the vertical position where the gradient ends, measured in a percentage of the icon height. Applied when the property Fill Type is set to gradient. Data type: Integer
Gradient Start Color	Specifies the color of the point where the gradient begins, applied when the property Fill Type is set to gradient. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to specify the color. Data type: String
Gradient Start X	Specifies the horizontal position where the gradient begins, measured in a percentage of the icon width. Applied when the property Fill Type is set to gradient. Data type: Integer
Gradient Start Y	Specifies the vertical position where the gradient begins, measured in a percentage of the icon height. Applied when the property Fill Type is set to gradient. Data type: Integer
Gradient Style	Specifies the gradient style for the icon, applied when the property Fill Type is set to gradient. Choose an option from the drop-down list. Data type: Enumeration
Height	Specifies the height of the icon, in inches. Enter a numeric value to change the width. Data type: Float
Icon Alignment	Specifies the relative position of the icon to the label. Choose an option from the drop-down list. Data type: Enumeration
Icon Style	Specifies the style of the icon attached to the label. Choose an option from the drop-down list. Data type: Enumeration
Icon Text Gap	Specifies the distance between the icon and the label. Data type: Float

Chart legend

The properties of a chart legend are:

Property	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. Enter a numeric value to change the position in inches. This property takes effect only when the Placement property is set to customized. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. Enter a numeric value to change the position in inches. This property takes effect only when the Placement property is set to customized. Data type: Float
Legend	
Auto Size	Specifies whether to resize the legend automatically. Data type: Boolean
Bottom Margin	Specifies the distance between the legend labels and the bottom border of the legend, in inches. Enter a numeric value to change the margin. Data type: Float
Label Format Source	Specifies the data format option for the chart legend labels. Choose an option from the drop-down list. • **Axis Data Format** - If selected, the data format of the legend labels will follow that of the axis data: if the chart has series axis, the legend label format will be the same as the series axis data format; if not, the category axis data format will be used to the legend labels. • **Legend Data Format** - If selected, the data format of the legend labels will be controlled by the property Legend Data Format. Unavailable to heat maps that are colored only by summaries. Data type: Enumeration
Label Horizontal Spacing	Specifies the horizontal distance between two adjacent legend labels. Enter a numeric value to change the distance. Unavailable to heat maps that are colored only by summaries. Data type: Float
Label Vertical Spacing	Specifies the vertical distance between two adjacent legend labels. Enter a numeric value to change the distance. Unavailable to heat maps that are colored only by summaries. Data type: Float
Left Margin	Specifies the distance between the legend labels and the left border of the legend, in inches. Enter a numeric value to change the margin. Data type: Float
Legend Data Format	Specifies the data format of the legend labels, applied when Label Format Source is set to Legend Data Format. Click ... in the value cell to specify the format in the Data Format dialog. Data type: String
Placement	Specifies the position of the legend in the platform. Choose an option from the drop-down list. Data type: Enumeration
Right Margin	Specifies the distance between the legend labels and the right border of the legend, in inches. Enter a numeric value to change the margin. Data type: Float
Secondary Placement	Specifies the position of the legend further based on the setting of the property Placement. Choose an option from the drop-down list. Data type: Enumeration
Show Scrollbar	Specifies whether to show a scrollbar on the legend to fully view the legend content when the content does not fit into the legend. Data type: Boolean
Show Tips	Specifies whether to show the corresponding data information when the mouse pointer points at a target in the chart legend in Designer view mode, in HTML result, or at server runtime. Data type: Boolean
Top Margin	Specifies the distance between the legend labels and the top border of the legend, in inches. Enter a numeric value to change the margin. Data type: Float
Truncate	Specifies whether to truncate the legend entry label text when the text overflow the labels. Data type: Boolean
Background	
Border Color	Specifies the color of the legend border. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border End Caps	Specifies the ending style of the legend border line. Choose an option from the drop-down list. Data type: Enumeration
Border Style	Specifies the line style of the legend border. Choose an option from the drop-down list. Data type: Enumeration
Border Thickness	Specifies the width of the legend border, in inches. Enter a numeric value to change the thickness. Data type: Float
Border Transparency	Specifies the transparency of the legend border, in percent. Enter a numeric value to change the transparency. Data type: Integer
Border Type	Specifies the border type of the legend. Choose an option from the drop-down list. Data type: Enumeration
Border Variable Dash	Specifies whether to resize the dash automatically. • **true** - If selected, the dash size will be adjusted automatically. • **false** - If selected, the dash size will be of fixed size. Data type: Boolean
Color	Specifies the color of the legend background, applied when the property Fill Type is set to color. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Fill Transparency	Specifies the transparency of the legend background, in percent. Enter a numeric value to change the transparency. Data type: Integer
Fill Type	Specifies the fill pattern for the legend background. Choose an option from the drop-down list. Data type: Enumeration
Inset Bottom	Specifies the bottom position of the background area, measured in a percentage of the legend height, from the bottom edge of the legend. Data type: Float
Inset Left	Specifies the left position of the background area, measured in a percentage of the legend width, from the left edge of the legend. Data type: Float
Inset Right	Specifies the right position of the background area, measured in a percentage of the legend width, from the right edge of the legend. Data type: Float
Inset Top	Specifies the top position of the background area, measured in a percentage of the legend height, from the top edge of the legend. Data type: Float
Label	
Color	Specifies the color of the label, applied when the property Fill Type is set to color. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Fill Transparency	Specifies the transparency of the background, in percent. Enter a numeric value to change the transparency. Data type: Integer
Fill Type	Specifies the fill pattern for the label. Choose an option from the drop-down list. Data type: Enumeration
Font Face	Specifies the font face for the legend labels. Choose an option from the drop-down list. Data type: Enumeration
Font Rotation	Specifies the rotation angle of each legend label around its center, in degrees. The default value is 0. Enter a numeric value to change the rotation. Data type: Float
Font Size	Specifies the font size for the legend labels. Enter an integer value to change the size. Data type: Integer
Font Style	Specifies the font style for the legend labels. Choose an option from the drop-down list. Data type: Enumeration
Outline Color	Specifies the color for the label text outline. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Outline End Caps	Specifies the ending style for the label text outline. Choose an option from the drop-down list. Data type: Enumeration
Outline Joint	Specifies the joint style for the label text outline. Choose an option from the drop-down list. Data type: Enumeration
Percent Format	Specifies the format of the percentage values on the legend. Applied when Show Percent is set to true. Click ... in the value cell to specify the format in the Data Format dialog. Data type: String
Reverse Labels	Specifies whether to reverse the order of the labels in the legend object. Data type: Boolean
Show Percent	Specifies whether to show the percentage of each entry value to the total on the legend. Applies to pie, bar and bench charts that have only one value axis. Data type: Boolean
Show Values	Specifies whether to show the values for the legend entries. Applies to pie, bar and bench charts that have only one value axis. Data type: Boolean
Text of Total Value	Specifies the text of the total value label on the legend. Applies to pie, bar and bench charts that have only one value axis. Data type: String
Value Format	Specifies the number format of the legend entry values. Applied when Show Values is set to true. Click ... in the value cell to specify the format in the Data Format dialog. Data type: String
Icon	
Border Color	Specifies the color of the icon borders. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border End Caps	Specifies the ending style of the icon border line. Choose an option from the drop-down list. Data type: Enumeration
Border Joint	Specifies the joint style of the icon border line. Choose an option from the drop-down list. Data type: Enumeration
Border Style	Specifies the line style of the icon borders. Choose an option from the drop-down list. Data type: Enumeration
Border Thickness	Specifies the width of the icon borders in inches. Enter a numeric value to change the thickness. Data type: Float
Border Transparency	Specifies the transparency of the icon borders, in percent. Enter a numeric value to change the transparency. Data type: Integer
Border Variable Dash	Specifies whether to resize the dash automatically. If the property is set to true, the dash size will be adjusted automatically; otherwise, the dash size will be of fixed size. Data type: Boolean
Height	Specifies the height of the icon, in inches. Enter a numeric value to change the width. Data type: Float
Icon Alignment	Specifies the relative position of the icon to the legend. Choose an option from the drop-down list. Data type: Enumeration
Icon Text Gap	Specifies the distance between each legend label and icon. Enter a numeric value to change the gap. Data type: Float

Chart paper

The properties of a chart paper are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. Enter a numeric value to change the position in inches. This property takes effect only when the Placement property for the chart legend is set to customized. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. Enter a numeric value to change the position in inches. This property takes effect only when the Placement property for the chart legend is set to customized. Data type: Float
Coordinate Paper	
Graph Position	Specifies the position of the paper contents relative to the paper background. Choose an option from the drop-down list. Data type: Enumeration
Reverse Painting Order	Specifies whether to reverse the order of graphic objects in a combo chart. Data type: Boolean
Scale X	Specifies the percentage for elongating or shortening the X axis. Enter a numeric value to change the percentage. Data type: Float
Scale Y	Specifies the percentage for elongating or shortening the Y axis. Enter a numeric value to change the percentage. Data type: Float
Background	
Border Color	Specifies the color of the paper border. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border End Caps	Specifies the ending style of the paper border line. Choose an option from the drop-down list. Data type: Enumeration
Border Joint	Specifies the joint style of the paper border line. Choose an option from the drop-down list. Data type: Enumeration
Border Style	Specifies the line style of the paper border. Choose an option from the drop-down list. Data type: Enumeration
Border Thickness	Specifies the width of the paper border, in inches. Enter a numeric value to change the thickness. Data type: Float
Border Transparency	Specifies the transparency of the paper border, in percent. Enter a numeric value to change the transparency. Data type: Integer
Border Type	Specifies the border type of the paper. Choose an option from the drop-down list. Data type: Enumeration
Border Variable Dash	Specifies whether to resize the dash automatically. • **true** - If selected, the dash size will be adjusted automatically. • **false** - If selected, the dash size will be of fixed size. Data type: Boolean
Color	Specifies the color of the paper background, applied when the property Fill Type is set to color. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog to specify the color and fill effect. Data type: String
Fill Transparency	Specifies the transparency of the paper background, in percent. Enter a numeric value to change the transparency. Data type: Integer
Fill Type	Specifies the fill pattern for the paper background. Choose an option from the drop-down list. Data type: Enumeration
Inset Bottom	Specifies the bottom position of the background area, measured in a percentage of the paper height, from the bottom edge of the paper. Data type: Float
Inset Left	Specifies the left position of the background area, measured in a percentage of the paper width, from the left edge of the paper. Data type: Float
Inset Right	Specifies the right position of the background area, measured in a percentage of the paper width, from the right edge of the paper. Data type: Float
Inset Top	Specifies the top position of the background area, measured in a percentage of the paper height, from the top edge of the paper. Data type: Float
Others (available to org charts only)	
Layout Mode	Specifies how the org chart tree expands from the root node. Choose an option from the drop-down list. • **top down** The org chart tree expands from top to bottom. • **bottom up** The org chart tree expands from bottom to top. It is the reverse of the top down mode. • **left right** The org chart tree expands from left to right. Data type: Enumeration
Graph (unavailable to heat map and org charts)	
Border	Specifies whether to show the borders. Data type: Boolean
Border Color	Specifies the line color for the border, not applying to 2-D line charts. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border End Caps	Specifies the ending style for the border, not applying to 2-D line charts. Choose an option from the drop-down list. Data type: Enumeration
Border Joint	Specifies the line joint style for the border, not applying to 2-D line charts. Choose an option from the drop-down list. Data type: Enumeration
Border Style	Specifies the line style for the border, not applying to 2-D line charts. Choose an option from the drop-down list. Data type: Enumeration
Border Thickness	Specifies the thickness for the border, not applying to 2-D line charts. Enter a numeric value to change the thickness. Data type: Float
Border Transparency	Specifies the line transparency for the border, in percent, not applying to 2-D line charts. Enter a numeric value to change the transparency. Data type: Integer
Suppress Label When 0	If true, the data label whose value is 0 will not be displayed. This property is available to these 2-D chart types: bar, bench, line, area and pie. Data type: Boolean
For bar/bench chart	
Bar Gap	Specifies the distance between each bar/bench in a category. Applies to 2-D clustered bar/bench charts only. Enter a numeric value to change the gap. Data type: Integer
Bar Style	Specifies the style of the bars/benches. Choose an option from the drop-down list. It can be normal or cylinder, however, if you specify the style to be cylinder, it will take effect only when the property Use Depth is set to true. Data type: Enumeration
Bar Width	Specifies the total width of the bars/benches in each category, measured in a percentage of the unit width. Enter a numeric value to change the width. Data type: Integer
Depth	Specifies the depth of the bars/benches, applied when the value of the property Use Depth is set to true. Applies to 2-D bar/bench charts only. Enter a numeric value to change the depth. Data type: Float
Direction	Specifies the angle of the axis along the depth of the bars/benches. Applies to 2-D bar/bench charts only. Enter a numeric value to change the direction. Data type: Integer
Update Direction	Specifies the direction in which to paint the values when the chart is updated. Available to real time chart only. Choose an option from the drop-down list. • **Default** - If it is a bench chart, paints the values from top to bottom, otherwise, from right to left. • **Reverse** - If it is a bench chart, paints the values from bottom to top, otherwise, from left to right. Data type: Enumeration
Use Depth	Specifies whether to make the bars/benches visually three-dimensional. Applies to 2-D bar/bench charts only. Data type: Boolean **Note:** If the chart is a combo chart composed by bars and areas/lines, when you set the depth properties for the bars, they will be applied to the areas/lines as well, and vice versa.
Value Label Type	Specifies in which way the value labels will be displayed around the bars/benches. Choose an option from the drop-down list. • **value** - Shows the value for each bar/bench. • **percent** - Shows the percentage of each bar/bench to the total. • **value and percent** - Shows the value and the percentage for each bar/bench. Data type: Enumeration
Vary Colors by Color List	Specifies whether or not to make the bar colors vary. It only takes effect on clustered bar/bench types and when the charts has no series field. Data type: Boolean
For line chart	
Area Pattern List	Specifies patterns for the areas which are formed by the chart axes and the lines. Applies to 2-D lines only. Click ... in the value cell and click the small squares in the color tray one by one to specify the patterns for the areas. The patterns can be one or more of the following: Automatic, Color, Texture, and Gradient. Data type: String
Depth	Specifies the depth of the lines, applied when the property Use Depth is set to true. Enter a numeric value to change the depth. Data type: Float
Direction	Specifies the angle of the axis along the depth of the lines. Enter a numeric value to change the direction. Data type: Integer
Ignore Null Value	Specifies whether to skip null data values in order to draw continuous lines. • **true** - When a null data value appears, it is skipped and the line is drawn from the previous data value to the next data value directly. • **false** - When a null data value appears, draw it anyway and the line is broken at the point of the null data value. Data type: Boolean
Update Direction	Specifies the direction in which to paint the values when the chart is updated. Available to real time chart only. Choose an option from the drop-down list. • **Default** - Paints the values from right to left. • **Reverse** - Paints the values from left to right. Data type: Enumeration
Use Depth	Specifies whether to make the lines visually three-dimensional. Applies to 2-D line charts only. Data type: Boolean **Note:** If the chart is a combo chart composed by lines and areas/bars, when you set the depth properties for the lines, they will be applied to the areas/bars as well, and vice versa.
Value Label Type	Specifies in which way the value labels will be displayed around the lines. Available to 2-D line charts only. Choose an option from the drop-down list. • **value** - Shows the value for each line node. • **percent** - Shows the percentage of each line node to the total. • **value and percent** - Shows the value and the percentage for each line node. Data type: Enumeration
For area chart	
Depth	Specifies the depth of the areas, applied when the property Use Depth is set to true. Applies to 2-D area charts only. Enter a numeric value to change the depth. Data type: Float
Direction	Specifies the angle of the axis along the depth of the areas. Applies to 2-D area charts only. Enter a numeric value to change the direction. Data type: Integer
Update Direction	Specifies the direction in which to paint the values when the chart is updated. Available to real time chart only. Choose an option from the drop-down list. • **Default** - Paints the values from right to left. • **Reverse** - Paints the values from left to right. Data type: Enumeration
Use Depth	Specifies whether to make the areas visually three-dimensional. Applies to 2-D area charts only. Data type: Boolean **Note:** If the chart is a combo chart composed by areas and bars/lines, when you set the depth properties for the areas, they will be applied to the bars/lines as well, and vice versa.
Use Dropline	Specifies whether to show the lines that represent the data categories. Applies to 2-D area charts only. Data type: Boolean
Value Label Type	Specifies in which way the value labels will be displayed around the areas. Choose an option from the drop-down list. • **value** - Shows the value for each area node. • **percent** - Shows the percentage of each area node to the total. • **value and percent** - Shows the value and the percentage for each area node. Data type: Enumeration
For pie chart	
Angle X	Specifies the rotation angle of the chart around the X axis, in degrees. Enter a numeric value to change the rotation. Data type: Float
Angle Y	Specifies the rotation angle of the chart around the Y axis, in degrees. Enter a numeric value to change the rotation. Data type: Float
Donut Hole	Specifies the radius percentage of the donut hole to the total pie circle. Applies to donut chart only. Data type: Float
Pie Gap	Specifies the distance between every two adjacent pie sections. Enter a numeric value to change the gap. Data type: Float
Section Explode Gap	Specifies the distance between each pie section and the pie center. Enter a numeric value to change the gap. Data type: Float
Show Pie/Donut Name	Specifies whether to show the name of the pie/donut. Data type: Boolean
Value Label Type	Specifies in which way the value labels will be displayed around the pie. Choose an option from the drop-down list. • **value** - Shows the value for the pie section. • **category name** - Shows the category name for the pie section. • **percent** - Shows the percentage of the section to the total. • **value and percent** - Shows the value and the percentage for the pie section. Data type: Enumeration
For indicator chart	
Horizontal Margin	Specifies the distance between the left indicators and the left border of the chart paper, in inches. Data type: Float
Range Radius	Specifies the relative size of an indicator in a percentage of the total indicator size. Data type: Integer
Vertical Margin	Specifies the distance between the top indicators and the top border of the chart paper, in inches. Data type: Float
For gauge chart	
Alert Color	Specifies the color for the alert range. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Alert Maximum Value	Specifies the maximum value for the alert range. Enter a numeric value to specify the value. Data type: Float
Alert Minimum Value	Specifies the minimum value for the alert range. Enter a numeric value to specify the value. Data type: Float
Arrow Style	Specifies the style for pointer arrows, applied when the property Pointer Style is set to Arrow. Applies to dial gauge charts only. Choose an option from the drop-down list. Data type: Enumeration
Curve Style	Specifies the style for the curve. Applies to dial gauge charts only. Choose an option from the drop-down list. Data type: Enumeration
Error Color	Specifies the color for the error range. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Error Maximum Value	Specifies the maximum value for the error range. Enter a numeric value to specify the value. Data type: Float
Error Minimum Value	Specifies the minimum value for the error range. Enter a numeric value to specify the value. Data type: Float
Fill Curve	Specifies whether to fill the dial with colors. Applies to dial gauge charts only. Data type: Boolean
Gauge Alert Type	Specifies the fill pattern for the alert range. Choose an option from the drop-down list. Data type: Enumeration
Gauge Error Type	Specifies the fill pattern for the error range. Choose an option from the drop-down list. Data type: Enumeration
Gauge Normal Type	Specifies the fill pattern for the normal range. Choose an option from the drop-down list. Data type: Enumeration
Normal Color	Specifies the color for the normal range. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Normal Maximum Value	Specifies the maximum value for the normal range. Enter a numeric value to specify the value. Data type: Float
Normal Minimum Value	Specifies the minimum value for the normal range. Enter a numeric value to specify the value. Data type: Float
Outer Inner Gap	Specifies the width of the dial. Applies to dial gauge charts only. Enter a numeric value to change the gap. Data type: Float
Pointer Style	Specifies the style for the pointer. Applies to dial gauge charts only. Choose an option from the drop-down list. Data type: Enumeration
Range Radius	Specifies the relative size of a dial or ball. Applies to dial and bubble gauge charts only. Enter a numeric value to change the size. Data type: Integer
Start Angle	Specifies the angle of the dial beginning point. Applies to dial gauge charts only. Enter a numeric value to change the angle. Data type: Integer
Stop Angle	Specifies the angle of the dial endpoint. Applies to dial gauge charts only. Enter a numeric value to change the angle. Data type: Integer
For bubble chart	
Cut Bubble Based on XY Area	Specifies whether to cut the bubbles when they are beyond the area formed by the X axis and Y axis. Data type: Boolean
Use 3D Effect	Specifies whether to make the bubbles visually three-dimensional. Data type: Boolean
For bullet chart	
Comparative Measure Width	Specifies the width of the comparative measures. Enter a numeric value to change the width. Data type: Integer
Featured Measure Width	Specifies the width of the featured measures. Enter a numeric value to change the width. Data type: Integer
Graph Direction	Specifies the direction to draw the bullet chart, horizontally or vertically. Choose an option from the drop-down list. Data type: Enumeration
Qualitative Ranges Width	Specifies the width of the qualitative ranges. Enter a numeric value to change the width. Data type: Integer
Special Featured Measure Graph Type	Specifies the graph type for the featured measures when the value (Y) axis does not start from the value zero. Choose an option from the drop-down list. Data type: Enumeration
Vary Colors by Color List	Specifies whether or not to make colors for the featured measures vary. Data type: Boolean
Hint (unavailable to org chart)	
Show Category and Series	Specifies whether to include the category and series values in the hints of the data markers. Data type: Boolean
Show Tips	Specifies whether to show the hint that displays the value of a data marker when the mouse pointer points at the data marker in HTML or Web Report result. If this property is set to false, all the other hint properties will be ignored. Data type: Boolean
Category (X) Axis (unavailable to heat map and org charts)	
Auto Major Unit	Specifies whether the interval unit between two adjacent major tick marks is defined by JReport automatically. If true, the unit will be defined by JReport automatically. If false, the user defined major interval unit will be used. Data type: Boolean
Auto Maximum	Specifies whether the maximum value which is used to label the tick marks is defined by JReport automatically. If true, the maximum value will be defined by JReport automatically. If false, the user defined maximum value will be used. Data type: Boolean
Auto Minor Unit	Specifies whether the interval unit between two adjacent minor tick marks is defined by JReport automatically. If true, the unit will be defined by JReport automatically. If false, the user defined minor interval unit will be used. Data type: Boolean
Auto Minimum	Specifies whether the minimum value which is used to label the tick marks is defined by JReport automatically. If true, the minimum value will be defined by JReport automatically. If false, the user defined minimum value will be used. Data type: Boolean
Axis Color	Specifies the color for the axis. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Axis End Caps	Specifies the ending style for the axis. Choose an option from the drop-down list. Data type: Enumeration
Axis Joint	Specifies the joint style for the axis. Choose an option from the drop-down list. Data type: Enumeration
Axis Style	Specifies the style for the axis. Choose an option from the drop-down list. Data type: Enumeration
Axis Thickness	Specifies the thickness for the axis. Enter a numeric value to change the thickness. Data type: Float
Axis Transparency	Specifies the transparency for the axis, in percent. Enter a numeric value to change the transparency. Data type: Integer
Axis Variable Dash	Specifies whether to resize the dash automatically. If the property is set to true, the dash size will be adjusted automatically; otherwise, the dash size will be of fixed size. Data type: Boolean
Gridline Color	Specifies the color for the gridlines. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Gridline End Caps	Specifies the ending style for the gridlines. Choose an option from the drop-down list. Data type: Enumeration
Gridline Joint	Specifies the joint style for the gridlines. Choose an option from the drop-down list. Data type: Enumeration
Gridline Style	Specifies the style for the gridlines. Choose an option from the drop-down list. Data type: Integer
Gridline Thickness	Specifies the thickness for the gridlines. Enter a numeric value to change the thickness. Data type: Float
Gridline Transparency	Specifies the transparency for the gridlines, in percent. Enter a numeric value to change the transparency. Data type: Integer
Gridline Variable Dash	Specifies whether to resize the dash automatically. If the property is set to true, the dash size will be adjusted automatically; otherwise, the dash size will be of fixed size. Data type: Boolean
Increment	Specifies the increased amount between two adjacent values on the axis. The default is to let the increased amount be determined by JReport Designer. Applies to bubble charts which use the axis to show numeric data only. Data type: Integer
Label Axis Gap	Specifies the distance between the labels and the axis. Enter a numeric value to change the gap. Data type: Float
Label Color	Specifies the color for the label text. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Label Every N Tick Marks	Specifies the frequency at which the tick marks will be labeled. Type 1 to label every tick mark, type 2 to label every two tick marks, type 3 to label every three tick marks, and so on. Data type: Integer
Label Fill Transparency	Specifies the transparency for the label text, in percent. Enter a numeric value to change the transparency. Data type: Integer
Label Fill Type	Specifies the fill pattern for the label text. Choose an option from the drop-down list. Data type: Enumeration
Label Font Face	Specifies the font face for the label text. Choose an option from the drop-down list. Data type: Enumeration
Label Font Orientation Automatically	Specifies whether to adjust the rotation angle of the label text on the axis automatically. • **true** - The rotation angle of the label text will be adjusted automatically according to the length of the text, in degrees. If the text can be completely displayed horizontally, the default rotation angle will be 0; if the text cannot be completely displayed horizontally, the default rotation angle will be 30 anticlockwise, and the cut off part will be shown as suspension points. • **false** - If selected, the rotation angle of the label text will be determined by value of the property Label Font Rotation. Data type: Boolean
Label Font Rotation	Specifies the rotation angle of the label text around its center, in degrees. The default value is 0. Enter a numeric value to change the rotation. Data type: Float
Label Font Size	Specifies the font size for the label text. Enter an integer value to change the size. Data type: Integer
Label Font Style	Specifies the font style for the label text. Choose an option from the drop-down list. Data type: Enumeration
Label Position	Specifies whether to show the label text outside or inside the axis. Takes effect only when Use Depth is set to false. Data type: Enumeration
Major Interval	Specifies the interval between two adjacent major tick marks. Enter a numeric value to change the interval. Data type: Float
Major Interval Unit	Specifies the interval unit between two adjacent major tick marks. Choose an option from the drop-down list. Data type: Enumeration
Minor Interval	Specifies the interval between two adjacent minor tick marks. Enter a numeric value to change the interval. Data type: Float
Minor Interval Unit	Specifies the interval unit between two adjacent minor tick marks. Choose an option from the drop-down list. Data type: Enumeration
Major Tick Mark Length	Specifies the length of the major tick marks. Enter a numeric value to change the length. Data type: Float
Major Tick Mark Line Color	Specifies the color of the major tick marks. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Major Tick Mark Line End Caps	Specifies the ending style for the major tick marks. Choose an option from the drop-down list. Data type: Enumeration
Major Tick Mark Line Joint	Specifies the joint style for the major tick marks. Choose an option from the drop-down list. Data type: Enumeration
Major Tick Mark Line Style	Specifies the line type for the major tick marks. Choose an option from the drop-down list. Data type: Enumeration
Major Tick Mark Line Thickness	Specifies the thickness of the major tick marks. Enter a numeric value to change the thickness. Data type: Float
Major Tick Mark Line Transparency	Specifies the transparency of the major tick marks, in percent. Enter a numeric value to change the transparency. Data type: Integer
Major Tick Mark Line Variable Dash	Specifies whether to adjust the dash size automatically. Applied when Major Tick Mark Line Style is not set to solid line. • **true** - The dash size will be adjusted automatically. • **false** - The dash size will be of fixed size. Data type: Boolean
Major Tick Mark Type	Specifies the position of the major tick marks relative to the axis. Choose an option from the drop-down list. Data type: Enumeration
Maximum	Specifies the maximum value which will be used to label the tick marks. Data type: Float
Maximum Value	Specifies the maximum value of the data appearing on the X axis. Applies to bubble charts which use the axis to show numeric data only. Data type: Float
Minimum	Specifies the minimum value which will be used to label the tick marks. Data type: Float
Minimum Value	Specifies the minimum value of the data appearing on the X axis. Applies to bubble charts which use the axis to show numeric data only. Data type: Float
Minor Tick Mark Length	Specifies the length of the minor tick marks. Enter a numeric value to change the length. Data type: Float
Minor Tick Mark Type	Specifies the position of the minor tick marks relative to the axis. Choose an option from the drop-down list. Data type: Enumeration
Number of Labels	Specifies the number of tick mark labels that will be shown. The default is to let JReport Designer determine the number of labels to be shown. Data type: Integer
Number of Tick Marks	Specifies the number of tick marks that will be shown. The default is to let the number be determined by JReport Designer. Applies to bubble charts only. Data type: Integer
Number of Visible Values	Specifies how many data items will be selected on the scrollbar and displayed on the axis by default. Available to 2-D scrollbar charts of bar, bench, line, area and stock types only. Data type: Integer
Percentage of Scrolling Area Height	Specifies the height percentage the scrollbar occupies the whole size of the chart. Available to 2-D scrollbar charts of bar, bench, line, area and stock types only. Data type: Float
Position Axis	Specifies the position at which the data point and label will be displayed on the X axis. • **on tick marks** - The data point and label will be aligned with each tick mark on the X axis. • **between tick marks** - The data point and label will be displayed between each two tick marks on the X axis. Data type: Enumeration
Scrollable Chart	Specifies whether to make the chart scrollable. If true, a scrollbar will be added in the chart, with which you can control the visible value range on the axis. Available to 2-D charts of bar, bench, line, and area types. Data type: Boolean
Show Axis	Specifies whether to show the axis. Data type: Boolean
Show Axis Label Tips	Specifies whether to show the complete label text when the mouse pointer points at a label on the axis in HTML or Web Report result. Data type: Boolean
Show Chart in Scrolling Area	Specifies whether to show the thumbnail chart on the scrollbar. Applied only when Scrollable Chart is set to true for 2-D charts of bar, bench, line and area types. Data type: Boolean
Show Gridline	Specifies whether to show the gridlines perpendicular to the axis. Data type: Boolean
Show Tick Mark Labels	Specifies whether to show the tick mark labels. Data type: Boolean
Use Best Effect	Specifies whether to hide the label which would overlap with the one in front of it. Data type: Boolean
Use Constant Interval	Specifies whether to use a constant interval to label the tick marks. If true, the values of the tick marks will be increased continually on the axis based on the following properties, instead of just using the data values. Data type: Boolean
Value (Y) Axis (unavailable to heat map and org charts)	
Auto Scale on Y Axis	Specifies whether to recalculate the scale of Value (Y) Axis when the the value of the property Number of Visible Values is changed. Applied only when Scrollable Chart is set to true for 2-D charts of bar, bench, line, and area types. Data type: Integer
Axis Color	Specifies the color for the axis. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Axis End Caps	Specifies the ending style for the axis. Choose an option from the drop-down list. Data type: Enumeration
Axis Joint	Specifies the joint style for the axis. Choose an option from the drop-down list. Data type: Enumeration
Axis Style	Specifies the style for the axis. Choose an option from the drop-down list. Data type: Enumeration
Axis Thickness	Specifies the thickness for the axis. Enter a numeric value to change the thickness. Data type: Float
Axis Transparency	Specifies the transparency for the axis, in percent. Enter a numeric value to change the transparency. Data type: Integer
Axis Variable Dash	Specifies whether to resize the dash automatically. If the property is set to true, the dash size will be adjusted automatically; otherwise, the dash size will be of fixed size. Data type: Boolean
Gridline Color	Specifies the color for the gridlines. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Gridline End Caps	Specifies the ending style for the gridlines. Choose an option from the drop-down list. Data type: Enumeration
Gridline Joint	Specifies the joint style for the gridlines. Choose an option from the drop-down list. Data type: Enumeration
Gridline Style	Specifies the style for the gridlines. Choose an option from the drop-down list. Data type: Enumeration
Gridline Thickness	Specifies the thickness for the gridlines. Enter a numeric value to change the thickness. Data type: Float
Gridline Transparency	Specifies the transparency for the gridlines, in percent. Enter a numeric value to change the transparency. Data type: Integer
Gridline Variable Dash	Specifies whether to resize the dash automatically. If the property is set to true, the dash size will be adjusted automatically; otherwise, the dash size will be of fixed size. Data type: Boolean
Increment	Specifies the increased amount between two adjacent values on the axis. The default is to let the increased amount be determined by JReport Designer. Data type: Integer
Label Axis Gap	Specifies the distance between the labels and the axis. Enter a numeric value to change the gap. Data type: Float
Label Color	Specifies the color for the label text. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Label Every N Tick Marks	Specifies the frequency at which the tick marks will be labeled. Type 1 to label every tick mark, type 2 to label every two tick marks, type 3 to label every three tick marks, and so on. Data type: Integer
Label Fill Transparency	Specifies the transparency for the label text, in percent. Enter a numeric value to change the transparency. Data type: Integer
Label Fill Type	Specifies the fill pattern for the label text. Choose an option from the drop-down list. Data type: Enumeration
Label Font Orientation Automatically	Specifies whether to adjust the rotation angle of the label text on the axis automatically. • **true** - The rotation angle of the label text will be adjusted automatically according to the length of the text, in degrees. If the text can be completely displayed horizontally, the default rotation angle will be 0; if the text cannot be completely displayed horizontally, the default rotation angle will be 30 anticlockwise, and the cut off part will be shown as suspension points. • **false** - If selected, the rotation angle of the label text will be determined by value of the property Label Font Rotation. Data type: Boolean
Label Font Face	Specifies the font face for the label text. Choose an option from the drop-down list. Data type: Enumeration
Label Font Rotation	Specifies the rotation angle of the label text around its center, in degrees. The default value is 0. Enter a numeric value to change the rotation. Data type: Float
Label Font Size	Specifies the font size for the label text. Enter an integer value to change the size. Data type: Integer
Label Font Style	Specifies the font style for the label text. Choose an option from the drop-down list. Data type: Enumeration
Label Position	Specifies whether to show the label text outside or inside the axis. Takes effect only when the property Use Depth is set to false. Data type: Enumeration
Major Tick Mark Length	Specifies the length of the major tick marks. Enter a numeric value to change the length. Data type: Float
Major Tick Mark Line Color	Specifies the color of the major tick marks. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Major Tick Mark Line End Caps	Specifies the ending style for the major tick marks. Choose an option from the drop-down list. Data type: Enumeration
Major Tick Mark Line Joint	Specifies the joint style for the major tick marks. Choose an option from the drop-down list. Data type: Enumeration
Major Tick Mark Line Style	Specifies the line type for the major tick marks. Choose an option from the drop-down list. Data type: Enumeration
Major Tick Mark Line Thickness	Specifies the thickness of the major tick marks. Enter a numeric value to change the thickness. Data type: Float
Major Tick Mark Line Transparency	Specifies the transparency of the major tick marks, in percent. Enter a numeric value to change the transparency. Data type: Integer
Major Tick Mark Line Variable Dash	Specifies whether to adjust the dash size automatically. Applied when Major Tick Mark Line Style is not set to solid line. • **true** - The dash size will be adjusted automatically. • **false** - The dash size will be of fixed size. Data type: Boolean
Major Tick Mark Type	Specifies the position of the major tick marks relative to the axis. Choose an option from the drop-down list. Data type: Enumeration
Maximum Value	Specifies the maximum value of the data appearing on the Y axis. Data type: Float
Minimum Value	Specifies the minimum value of the data appearing on the Y axis. Data type: Float
Minor Tick Mark Length	Specifies the length of the minor tick marks. Enter a numeric value to change the length. Data type: Float
Minor Tick Mark Type	Specifies the position of the minor tick marks relative to the axis. Choose an option from the drop-down list. Data type: Enumeration
Number of Labels	Specifies the number of tick mark labels that will be shown. The default is to let JReport Designer determine the number of labels to be shown. Data type: Integer
Number of Tick Marks	Specifies the number of tick marks that will be shown. The default is to let the number be determined by JReport Designer. Data type: Integer
Position Axis	Specifies the position at which the data point and label will be displayed on the X axis. • **on tick marks** - The data point and label will be aligned with each tick mark on the X axis. • **between tick marks** - The data point and label will be displayed between each two tick marks on the X axis. Data type: Enumeration
Show Axis	Specifies whether to show the axis. Data type: Boolean
Show Axis Label Tips	Specifies whether to show the complete label text when the mouse pointer points at a label on the axis in HTML or Web Report result. Data type: Boolean
Show Gridline	Specifies whether to show the gridlines perpendicular to the axis. Data type: Boolean
Show Tick Mark Labels	Specifies whether to show the tick mark labels. Data type: Boolean
Start Value	Specifies the position where the axis label starts to be shown, applied when the property Use Start Value is set to true. Applies to 2-D charts that have a wall only. Data type: Float
Use Best Effect	Specifies whether to hide the label which would overlap with the one in front of it. Data type: Boolean
Use Start Value	Specifies whether to set the start value of the axis. Applies to 2-D charts that have a wall only. Data type: Boolean
Wall (unavailable to gauge, heat map and org chart)	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border End Caps	Specifies the ending style of the wall border line. Choose an option from the drop-down list. Data type: Enumeration
Border Joint	Specifies the joint style of the wall border line. Choose an option from the drop-down list. Data type: Enumeration
Border Style	Specifies the line style of the wall borders. Choose an option from the drop-down list. Data type: Enumeration
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Border Transparency	Specifies the transparency of the wall borders, in percent. Enter a numeric value to change the transparency. Data type: Integer
Border Variable Dash	Specifies whether to resize the dash automatically. If the property is set to true, the dash size will be adjusted automatically; otherwise, the dash size will be of fixed size. Data type: Boolean
Bound Color	Specifies the color of the bound wall, applied when the property Bound Fill Type is set to color. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Bound Fill Transparency	Specifies the transparency of the color schema to compound with the basic fill which is set in the wall area, in percent. Enter a numeric value to change the transparency. Data type: Integer
Bound Fill Type	Specifies the fill type of the wall. Choose an option from the drop-down list. Data type: Enumeration
Color	Specifies the color of the wall, applied when the property Fill Type is set to color. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Fill Transparency	Specifies the transparency of the background, in percent. Enter a numeric value to change the transparency. Data type: Integer
Fill Type	Specifies the fill pattern for the wall. Choose an option from the drop-down list. Data type: Enumeration
Show Wall	Specifies whether to show the wall. Data type: Boolean
Threshold Line (unavailable to gauge, bubble, heat map and org charts)	
Fill Threshold1	Specifies whether to fill the first threshold area with a semi-transparent color. Data type: Boolean
Fill Threshold2	Specifies whether to fill the second threshold area with a semi-transparent color. Data type: Boolean
Fill Transparency	Specifies the transparency of the background, in percent. Enter a numeric value to change the transparency. Data type: Integer
Show Threshold Line1	Specifies whether to show the first threshold line. Data type: Boolean
Show Threshold Line2	Specifies whether to show the second threshold line. Data type: Boolean
Threshold Line Color1	Specifies the color for the first threshold line. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Threshold Line Color2	Specifies the color for the second threshold line. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Threshold Line Style	Specifies the style for the threshold lines. Choose an option from the drop-down list. Data type: Enumeration
Threshold Value1	Specifies the value for the first threshold line. Data type: Float
Threshold Value2	Specifies the value for the second threshold line. Data type: Float
MeanLine (unavailable to gauge, bubble, heat map and org charts)	
Color	Specifies the color of the mean line. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Node Style	Specifies the style of the nodes that represent the average values. Choose an option from the drop-down list. Data type: Enumeration
Show	Specifies whether to show a line that represents the average value of each category group. Not applies to bubble charts. Data type: Boolean
Type	Specifies the type of the mean line. Choose an option from the drop-down list. Data type: Enumeration

Notes:

- For combo chart, the graph properties will respectively apply to the corresponding chart types.
- A combo chart may contain two value axes, namely, the Y1 and Y2 axes. The properties of the Y1 or Y2 axis are the same as those of the Y axis.

The following are the properties of the two child objects of the chart paper in a heat map:

- Heat map rectangle
- Heat map rectangle title

The following are the properties of the two child objects of the chart paper in an org chart:

- Org chart line
- Org chart node

Heat map rectangle

The properties of the rectangles in a heat map are:

Property	Description
Others	
Vertical Alignment	Specifies the vertical justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Separator	
Color	Specifies the color of the separator line. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Line Style	Specifies the line style of separator. Choose an option from the drop-down list. Data type: Enumeration
Thickness	Specifies the line width of the separator, in inches. Enter a numeric value to change the thickness. Data type: Float
Transparency	Specifies the transparency of separators, in percent. Enter a numeric value to change the transparency. Data type: Integer

Heat map rectangle title

The properties of the rectangle titles in a heat map are:

General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Text Format	
Auto Fit	Specifies whether to adjust the width of the object according to the contents. Data type: Boolean
Bold	Specifies whether to make the text bold. Data type: Boolean
Font Face	Specifies the font of the text. Choose an option from the drop-down list. Data type: Enumeration
Font Size	Specifies the font size of the text. Enter an integer value to change the size. Data type: Integer
Horizontal Alignment	Specifies the horizontal justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Italic	Specifies whether to make the text italic. Data type: Boolean
Strikethrough	Specifies whether to add a strikethrough line to the text. Data type: Boolean
Underline	Specifies whether to underline the text. Data type: Boolean
Vertical Alignment	Specifies the vertical justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Word Wrap	Specifies whether to wrap the text to the width. Data type: Boolean
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Enabled when it is a horizontal banded object. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Foreground	Specifies the foreground color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Padding	
Bottom Padding	Specifies the space between the text and the bottom border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Left Padding	Specifies the space between the text and the left border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Right Padding	Specifies the space between the text and the right border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Top Padding	Specifies the space between the text and the top border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration
Others	
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. Data type: Boolean

Org chart line

The properties of the line object in an org chart are:

Property Name	Description
General	
Class Type	Indicates the class type of the object. This property is read only.
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Line Property	
Line Color	Specifies the color of the line. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Line Style	Specifies the style of the line. Choose an option from the drop-down list. Data type: Enumeration
Line Thickness	Specifies the thickness of the line. Enter a numeric value to change the thickness. Data type: Float

Org chart node

The properties of each node in an org chart are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Others	
Minimum Leaf Distance	Specifies the minimum distance between two adjacent leaf nodes on the same level. For top-down and bottom-up layouts, it means horizontal distance. For left-right layout, it means vertical distance. Enter a numeric value to specify the distance. Data type: Float
Minimum Level Distance	Specifies the minimum distance between two adjacent levels. For top-down and bottom-up layouts, it means vertical distance. For left-right layout, it means horizontal distance. When running the report in Web Report format, if the vlaue of Minimum Level Distance is less than that of Line Thickness on org chart line, the former will be overridden by the latter. Data type: Float
Vertical Alignment	Specifies the vertical justification of the objects in each node. Choose an option from the drop-down list. Data type: Enumeration
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to specify the color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration

Chart platform

A chart platform is a container for other chart elements. You can use the properties listed in the following table to control the appearances or behavior of a chart platform. In the Inspector panel, a chart platform is displayed as ChartObject by default.

The properties of a chart platform are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Others	
Export to Applet	Specifies whether to include the object when exporting the report to Applet. Data type: Boolean
Export to Excel	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
Export to HTML	Specifies whether to include the object when exporting the report to HTML. Data type: Boolean
Export to PDF	Specifies whether to include the object when exporting the report to PDF. Data type: Boolean
Export to PostScript	Specifies whether to include the object when exporting the report to PostScript. Data type: Boolean
Export to Report Result	Specifies whether to include the object when exporting the report to Report Result. Data type: Boolean
Export to RTF	Specifies whether to include the object when exporting the report to RTF. Data type: Boolean
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. You can also use a formula to control whether or not to show the object. Data type: Boolean
Position	Specifies the position of the object. Available when the object is directly contained in the report body or a tabular cell. • **absolute** - The object's position will be decided by its X and Y property values. • **static** - The object will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled. Data type: Enumeration
Platform	
Minimum Tick Mark Space	Specifies the minimum space between two adjacent tick marks. Enter a numeric value to change the space. Data type: Float
Show Legend	Specifies whether to make the legend visible in a chart. Data type: Boolean
Background	
Border Color	Specifies the color of the platform border. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border End Caps	Specifies the ending style of the platform border line. Choose an option from the drop-down list. Data type: Enumeration
Border Joint	Specifies the joint style of the platform border line. Choose an option from the drop-down list. Data type: Enumeration
Border Style	Specifies the line style of the platform border. Choose an option from the drop-down list. Data type: Enumeration
Border Thickness	Specifies the width of the platform border, in inches. Enter a numeric value to change the thickness. Data type: Float
Border Transparency	Specifies the transparency of the platform border, in percent. Enter a numeric value to change the transparency. Data type: Integer
Border Type	Specifies the border type of the platform. Choose an option from the drop-down list. Data type: Enumeration
Border Variable Dash	Specifies whether to resize the dash automatically. • **true** - If selected, the dash size will be adjusted automatically. • **false** - If selected, the dash size will be of fixed size. Data type: Boolean
Color	Specifies the color of the platform background, applied when the property Fill Type is set to color. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog to specify the color and fill effect. Data type: String
Fill Transparency	Specifies the transparency of the platform background, in percent. Enter a numeric value to change the transparency. Data type: Integer
Fill Type	Specifies the fill pattern for the platform background. Choose an option from the drop-down list. Data type: Enumeration
Inset Bottom	Specifies the bottom position of the background area, measured in a percentage of the chart height, from the bottom edge of the chart. Data type: Float
Inset Left	Specifies the left position of the background area, measured in a percentage of the chart width, from the left edge of the chart. Data type: Float
Inset Right	Specifies the right position of the background area, measured in a percentage of the chart width, from the right edge of the chart. Data type: Float
Inset Top	Specifies the top position of the background area, measured in a percentage of the chart height, from the top edge of the chart. Data type: Float
Radius	Specifies the radius for the joint of the platform border line. The property takes effect only when Border Joint is set to joint round. Enter a numeric value to change the radius. Data type: Float
Data	
Category End Offset	The four properties work together to control the range of the data being displayed on a chart: Category Start Offset, Category End Offset, Series Start Offset, and Series End Offset. Category End Offset specifies the ending offset of the categories. It does not apply to the charts which involve only one group. Data type: Integer
Category Format	Specifies the data format for the category axis (the X axis) to display the tick mark labels in the way you choose. Click ... in the value cell to specify the format in the Data Format dialog. Data type: String
Category Start Offset	The four properties work together to control the range of the data being displayed on a chart: Category Start Offset, Category End Offset, Series Start Offset, and Series End Offset. Category Start Offset specifies the starting offset of the categories. It does not apply to the charts which involve only one group. Data type: Integer
Hint Percent Format	Specifies the format for each hint percentage to the total. Click ... in the value cell to specify the format in the Data Format dialog. Data type: String
Hint Value Format	Specifies the number format for the hint message. Click ... in the value cell to specify the format in the Data Format dialog. Data type: String
Hyperlink	Specifies whether to add a hyperlink that refers to another report or a website to the data markers of the chart. Data type: String
Hyperlink Target	Specifies the target window or frame in which to display the content the Hyperlink property specifies. Choose an option from the drop-down list. • **<Server Setting>** Loads the linked file according to setting of the Pop Up New Window for Links option in the Customize Profile panel of the Profile dialog in JReport Server. • **New Window** Loads the linked file into a new window. The window is not named. • **Whole Window** Loads the linked file into the full browser window. • **Same Frame** Loads the linked file into the same frame as the link. • **Parent Frame** Loads the linked file into the parent frame of the frame that contains the link. • **Other Frame** Loads the linked file into some other specified frame. If selected, you can also directly input the name of the frame you have defined in the text box. If the frame name does not exist, the linked file will be loaded into a new window. Data type: String
Motion Bar Format	Specifies the data format for the motion field values displayed on the motion bar. Click ... in the value cell to specify the format in the Data Format dialog. Data type: String
Overall Series	Specifies whether to calculate the top or bottom N category values based on the series values. Data type: Boolean
Primary Data Format	Specifies the data format for the primary value axis (the Y1 axis) to display the tick mark labels in the way you choose. Click ... in the value cell to specify the format in the Data Format dialog. Data type: String
Reverse Category	Specifies whether to reverse the sequence of the category field value. Data type: Boolean
Reverse Series	Specifies whether to reverse the sequence of the series field value. Data type: Boolean
Secondary Data Format	Specifies the data format for the secondary value axis (the Y2 axis) to display the tick mark labels in the way you choose. Click ... in the value cell to specify the format in the Data Format dialog. Data type: String
Series End Offset	The four properties work together to control the range of the data being displayed on a chart: Category Start Offset, Category End Offset, Series Start Offset, and Series End Offset. Series End Offset specifies the ending offset of the series. Data type: Integer
Series Format	Specifies the data format for the series axis (the Z axis) to display the tick mark labels in the way you choose. Click ... in the value cell to specify the format in the Data Format dialog. Data type: String
Series Start Offset	The four properties work together to control the range of the data being displayed on a chart: Category Start Offset, Category End Offset, Series Start Offset, and Series End Offset. Series Start Offset specifies the starting offset of the series. Data type: Integer
Sort Category	Specifies the sorting order for the category field values. Choose an option from the drop-down list. Data type: Enumeration
Sort Series	Specifies the sorting order for the series field values. Choose an option from the drop-down list. Data type: Enumeration
Swap Groups	Specifies to display values from different data fields by switching data between the category and series axes, the category and value axes of a chart. By default, this property is set to false, which means no switch will take place in the chart. • **Switching between the category and series axes** If you create a chart by adding fields to all of its axes, and set Swap Groups to true, data on the category and series axes will be switched. • **Switching between the category and value axes** If you create a chart by adding one field to the category axis and several to the value axis, and set Swap Groups to true, data on the category and value axes will be switched. Data type: Boolean **Note:** If the chart has only one field on the category axis and one on the value axis, even though the property value is set to true, there will be no switch take place on the two axes.
X HyperLink	Specifies whether to add a hyperlink that refers to another report or a website to the X axis labels. Data type: String
X Hyperlink Target	Specifies the target window or frame in which to display the content the X Hyperlink property specifies. Choose an option from the drop-down list. • **<Server Setting>** Loads the linked file according to setting of the Pop Up New Window for Links option in the Customize Profile panel of the Profile dialog in JReport Server. • **New Window** Loads the linked file into a new window. The window is not named. • **Whole Window** Loads the linked file into the full browser window. • **Same Frame** Loads the linked file into the same frame as the link. • **Parent Frame** Loads the linked file into the parent frame of the frame that contains the link. • **Other Frame** Loads the linked file into some other specified frame. If selected, you can also directly input the name of the frame you have defined in the text box. If the frame name does not exist, the linked file will be loaded into a new window. Data type: String

•

A detailed chart property reference

Border Type

Specifies the border type of an object. The settings for Border Type are:

- **none**
  The object has no visible border lines.
- **raised**
  The object has 3-D borders that appear as if they are raised off the page.
- **recess**
  The object has 3-D borders that appear as if they are pressed into the page.
- **shadow**
  The object has two shadowed borders, beneath and to the right of the object.
- **solid**
  The object has single-line borders.

Data Format

Specifies the data format for the labels of the axes or legend as required.

Format Type	Option	Description (Sample)
Scale		Obtains a new value by a calculation on the value.
	Logarithm	Calculates ten to the power of the value.
	Hundreds	Divides the value by one hundred.
	Thousands	Divides the value by one thousand.
	Millions	Divides the value by one million.
	Billions	Divides the value by one billion.
	Trillions	Divides the value by one trillion.
Number		Re-formats the number value (original example: 123456)
	0	Formats a decimal number to an integer (123456).
	#,##0	Formats a decimal number to a digit grouped integer (123,456).
	#,##0;-#,##0	Formats a decimal number to a digit grouped integer. A minus sign is used as the negative prefix (123,456/-123,456).
	0.00	Formats a decimal number to a fixed-point number retaining 2 digits after decimal separator (123456.00).
	#,##0.00	Formats a decimal number to a digit grouped number retaining 2 digits after decimal separator (123,456.00).
	#,##0.00;-#,##0.00	Formats a decimal number to a digit grouped number retaining 2 digits after decimal separator. A minus sign is used as the negative prefix (123,456.00/-123,456.00).
	0.00E00	Formats a decimal number to a number in scientific notation. The mantissa is often in the range 1.0 <= x < 10.0, and the number of digit characters after the exponent character (E) gives the minimum exponent digit count (1.23E04).
	##0.0E0	Formats a decimal number to a number in scientific notation. The minimum number of integer in the mantissa is 1, and the number of digit characters after the exponent character (E) gives the minimum exponent digit count (12.34E3).
	$0	Format a decimal number to an integer, prefixed with a currency symbol $ ($123456).
	$#,##0	Formats a decimal number to a digit grouped integer, prefixed with a currency symbol $ ($123,456).
	$#,##0;-$#,##0	Formats a decimal number to a digit grouped integer, prefixed with a currency symbol $. A minus sign is used as the negative prefix ($123,456/-$123,456).
	$0.00	Formats a decimal number to a fixed-point number retaining 2 digits after decimal separator, prefixed with a currency symbol $ ($123456.00).
	$#,##0.00	Formats a decimal number to a digit grouped number retaining 2 digits after decimal separator, prefixed with a currency symbol $ ($123,456.00).
	$#,##0.00;-$#,##0.00	Formats a decimal number to a digit grouped number retaining 2 digits after decimal separator, prefixed with a currency symbol $. A minus sign is used as the negative prefix ($123,456.00/-$123,456.00).
	$0.00E00	Formats a decimal number to a number in scientific notation, prefixed with a currency symbol $. The mantissa is often in the range 1.0 <= x < 10.0, and the number of digit characters after the exponent character (E) gives the minimum exponent digit count ($1.23E$04).
	$##0.0E0	Formats a decimal number to a number in scientific notation, prefixed with a currency symbol $. The minimum number of integer in the mantissa is 1, and the number of digit characters after the exponent character (E) gives the minimum exponent digit count ($12.34E$3).
	0%	Formats a decimal number to percentage: multiplied by 100 and shown as an integer percentage (12%).
	0.00%	Formats a decimal number to percentage: multiplied by 100 and shown as a fixed-point number percentage retaining 2 digits after decimal separator (12.34%).
	Notes: • The number of 0 that appears in a pattern indicates the minimum digits, show 0 if that digit is zero. The number of # that appears after the decimal point in a pattern indicates the maximum digits of decimal, zero shows as absent. • Use ' (single quotation mark) to quote special characters in a prefix or suffix, for example, '#'# formats 123 to #123. To create the single quote itself, use two in a row: # o''clock.	
Date/Time		Re-formats the date/time value (For example, Wednesday, December 25, 00:00:00 GMT-08:00 2002)
	G	Formats a date to era designator (AD).
	yyyy	Formats a date to year (2002).
	yy	Formats a date to year in short form (02).
	yyyy G	Formats a date to year, tagged with era designator (2002 AD)
	MM	Formats a date to month in year, shown in number format (12).
	MMM	Formats a date to month in year, shown in the abbreviated form (Dec).
	MMMMMMMM	Formats a date to month in year, shown in the full form (December).
	dd/MM/yy	Formats a date to a simple date form (25/12/02).
	dd-MMM-yy	Formats a date to a simple date form (25-Dec-02).
	dd-MMM	Formats a date to a simple date form (25-Dec).
	MMM-yy	Formats a date to a simple date form (Dec-02).
	MMM yyyy	Formats a date to a simple date form (Dec 2002).
	dd	Formats a date to day in month (25).
	DDD	Formats a date to day in year (359).
	ww	Formats a date to week in year (52).
	W	Formats a date to week in month (4).
	EEE	Formats a date to day in week, in short form (Wed).
	EEEEEE	Formats a date to day in week (Wednesday).
	HH	Formats a date to hour in day, 0~23 (00).
	kk	Formats a date to hour in day, 1~24 (24).
	KK a	Formats a date to hour in am/pm, 0~11, with am/pm marker (00 AM)
	hh a	Formats a date to hour in am/pm, 1~12, with am/pm marker (12 AM)
	mm	Formats a date to minute in hour (00).
	hh:mm a	Formats a date to hour in am/pm, 1~12, minutes in an hour, with am/pm marker (12:00 AM).
	ss	Formats a date to second in minute (00).
	hh:mm:ss	Formats a date to hour in am/pm 1~12, minutes in an hour, and second in minute (12:00:00).
	hh:mm:ss a	Formats a date to hour in am/pm 1~12, minutes in an hour, and second in minute, with am/pm marker (12:00:00 AM).
	mm:ss	Formats a date to minute in hour and second in minute (00:00).
	MMMMM dd yyyy G (EEEEEE) hh:mm:ss aa z	Formats a date to a full form date, which contains month in year (shown in full form), days in a month, year, era designator, day in week (shown in the full form), hour in a day, 1~12, minutes in an hour, second in minute, am/pm marker, and time zone (December 25 2002 AD (Wednesday) 12:00:00 AM GMT-08:00).
	Notes: • Any characters in the pattern that are not in the ranges of ['a'..'z'] and ['A'..'Z'] will be treated as quoted text. For instance, characters such as ':', '.', ' ', '#' and '@' will appear in the resulting time text even they are not embraced within single quotes. • A pattern containing any invalid pattern letter cannot be added to the filter stack.	
Text		Specifies the length of the value.
	Default Length	Uses the length of the string Default Length, which is 14, as the length of the string. Any letters that exceed this length will be cut. You can modify this string, adding or deleting letters to increase or decrease the length limit. For example, type Teddy, the label displayed may probably be Wed D, which originally could be Wed December 25 2002. If you want to show all, you can type Teddy is a lovely bear!, where the length is longer than the length of string Wed December 25 2002.
Mapping		Maps new value to one or more values.
	One-to-one Mapping	Maps a new value to one value. Type the argument number that you want to replace, and then specify the new value in the Map to box. For example, if you want to replace the second data label on the X axis, which is Thu Jul 04 2002, with a new string US National Day, just type 2 in the Argument# box, type US National Day in the Map to box, and then click Add to add it to the Filter Stack. If you want to map more values, repeat this procedure.
	Range Mapping	Maps a new value to a range of values. Type the argument numbers to define the range you want to map together, and then specify the new value in the Map to box. For example, if you want to replace the data labels from the second one to the fifth with a new name My Holiday, just type 2 in the first Argument# box and 5 in the second Argument# box, then type My Holiday in the Map to box, and finally click Add to add it to the Stack. If you want to map ranges that are not consecutive, repeat this procedure.
	Text Mapping	Changes the data in the axis data label to a customized string. For example, if you want to change Chen in the first data label of the X axis to Jinfonet, type Chen in the Original Text box, type Jinfonet in the Map to box, and then click Add to add it to the Stack. You can only change the data one by one. Make sure that your spelling in the Original Text box is exactly the same as the data shown in the axis label.
	Prefix	Adds a string before the text in the data label of the axis. For example, if you want to add NA before the text in the data label of the X axis, type NA in the Map to box, and click Add to add it to the Stack.
	Suffix	Adds a string behind the text in the data label of the axis. For example, if you want to add NA behind the text in the data label of the X axis, type NA in the Map to box, and click Add to add it to the Stack.
	Note: The data labels along each axis count from 0, which means the number of the first data label you see on the axis will be 0, the second will be 1, the third will be 2, and so on.	

End Caps

Specifies the ending style of the border line. Can be one of the following:

- **butt**
  Ends unclosed subpaths and dash segments with no added decoration.
- **round**
  Ends unclosed subpaths and dash segments with a round decoration that has a radius equal to half of the width of the pen.
- **square**
  Ends unclosed subpaths and dash segments with a square projection that extends beyond the end of the segment to a distance equal to half of the line width.

Fill Type

Specifies the fill pattern of an object. Applies to chart platforms, chart paper, chart legends, walls, floors, axes, tick marks, labels, fonts and icons. Can be one of the following:

- **None**
  No fill (default).
- **Color**
  Fills with a specified color.
- **More Effects**
  Specifies either a gradient or an image as the fill effect in the Fill Effects dialog.

Border Joint

Specifies the border joint style. Can be one of the following:

- **miter**
  Joins path segments by extending their outside edges until they meet.
- **round**
  Joins path segments by rounding off the corner at a radius of half the line width.
- **bevel**
  Joins path segments by connecting the outer corners of their wide outlines with a straight segment.
- **joint round**
  Joins path segments by rounding off the corner at the specified radius. Available only for chart platform.

Start Offset(1st.Data.Set), End Offset(1st.Data.Set), Start Offset(2nd.Data.Set), and End Offset(2nd.Data.Set)

These four properties are used to control the data source range that appears on the chart.

For 2-level-group charts, record-level charts and all kinds of combination charts, Start Offset(1st.Data.Set) and End Offset(1st.Data.Set) are used to control the starting offset and ending offset of the data series; Start Offset(2ndDataSet) and End Offset(2nd Data.Set) are used to control the range of the categories.

For 1-level-group charts, Start Offset(2nd.Data.Set) and End Offset(2nd.Data.Set) are used to control the starting offset and ending offset of the categories; Start Offset(1st.Data.Set) and End Offset(1st.Data.Set) will not work because there is no series data in chart that contains only one group.

The range of the property values can be -1 (Default, Not Set) or an integer between 0 and Number of Data Series – 1 (or, Number of Categories - 1).

If the value is out of this range, the following rules will be applied to the default value:

IF	THEN
Start Offset(1st.Data.Set) < 0	Start Offset(1st.Data.Set) = 0
Start Offset(1st.Data.Set) > max data number	Start Offset(1st.Data.Set) = max data number
End Offset(1st.Data.Set) < 0	End Offset(1st.Data.Set) = max data number
End Offset(1st.Data.Set) > max data number	End Offset(1st.Data.Set) = max data number
Start Offset(1st.Data.Set) > End Offset(1st.Data.Set) and EndOffset(1st.Data.Set) >=0	Start Offset(1st.Data.Set) = 0, End Offset (1st.Data.Set) = max data number

Same rules apply to Start Offset(2nd.Data.Set) and End Offset(2nd.Data.Set).

Threshold Line Style

Specifies the style for the threshold lines. Applies to chart paper. Note that Stock Charts, Radar Charts, Scatter Charts, and Bubble Charts have no threshold lines.

A threshold line marks a specific data point that is specified by the user. It is usually used by the user for comparing with data series to see whether the data is higher or lower than this point. For example, if you want to see whether the production zones successfully accomplished their production tasks, you can set a threshold line to represent the goal output, and another to represent the lowest acceptable output quantity. By using the threshold lines, you can spot out the zones that are out of the range at a glance.

For 3-D chart, the threshold line value is represented by a plane.

Can be one of the following:

Option	Description
Style1	Use two threshold areas (Threshold Line1 and Threshold Line2) to emphasize the data series that are higher than the first threshold line or lower than the second threshold line.
Style2	Use a single threshold area (Threshold Line1) to emphasize the data series that are between the higher and the lower threshold lines.

Hyperlink

This property is used to add a hyperlink, which refers to another report or a website, to the chart data markers. You can control the hyperlink property with a formula, which is a good way to get required data from another report.

For example, you have two reports A and B in the SampleReports.cat catalog. In report A, the data in a table is grouped by customer country and a bar chart is used to illustrate the count of Customer_ID in every group. Report B shows the information of customer, such as ID, Name, and Annual Sales. Now we build a link between the two reports. Run report A, click the bar of a country, and report B appears, only showing the data of the specific country. The steps are:

1. Run report A in Web Report Studio.
2. Create a new formula named Link_b. It might be:
   ```
   string t="http://localhost:8888/jrserver/test/SampleReports.cat/ReportB.cls?jrs.cmd=jrs.web_vw&jrs.authorization=YWRtaW46YWRtaW4%3D&";
   string t1="jrs.param$PCOUNTRY=";
   string url=t+t1+"@INNER"+"&jrs.result_type=1";
   return url
   ```

   Notes:

   - Make sure that the paths of the catalog and the report in the URL are corresponding with their paths on the server.
   - The built-in parameter INNER is used for representing the category data. In addition, you can use OUTER for the series data.
   - For the X HyperLink property, you can use @XDIM and @ZDIM respectively instead of @INNER and @OUTER.
3. In the Inspector panel, set the Hyperlink property of the chart platform to **Link_b**.
4. Point to a bar and click it, the corresponding records will be displayed. You can also view the information of the other countries by clicking the corresponding bars in report A.

Crosstab

The properties of a crosstab in a web report are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Alternating Line Color	
Direction	Specifies whether the color defined by the property Pattern List will apply to the fields in the same rows or columns in the crosstab. • **vertical** - The value of the property Pattern List will take effect on the fields in the same columns, except for the total columns. • **horizontal** - The value of the property Pattern List will take effect on the fields in the same rows, except for the total rows. Data type: Enumeration
Enable	Specifies whether to enable setting alternating line color for the crosstab. Only when it is set to true, the setting for the property Direction can take effect. Data type: Boolean **Note:** The setting of the alternating line color function for crosstab can only take effect when all the aggregate fields in the crosstab have transparent background.
Others	
Enable Navigation	Specifies whether to enable scrollbars on the crosstab when the parent container cannot fully display the data of the crosstab. When this property is set to true, you can further specify the height and width of the frame for holding the crosstab, by setting the two properties Navigation Height and Navigation Width. Data type: Boolean
Export to Applet	Specifies whether to include the object when exporting the report to Applet. Data type: Boolean
Export to CSV	Specifies whether to include the object when exporting the report to CSV. Data type: Boolean
Export to Excel	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
Export to HTML	Specifies whether to include the object when exporting the report to HTML. Data type: Boolean
Export to PDF	Specifies whether to include the object when exporting the report to PDF. Data type: Boolean
Export to PostScript	Specifies whether to include the object when exporting the report to PostScript. Data type: Boolean
Export to Report Result	Specifies whether to include the object when exporting the report to Report Result. Data type: Boolean
Export to RTF	Specifies whether to include the object when exporting the report to RTF. Data type: Boolean
Export to Text	Specifies whether to include the object when exporting the report to Text. Data type: Boolean
Export to XML	Specifies whether to include the object when exporting the report to XML. Data type: Boolean
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. You can also use a formula to control whether or not to show the object. Data type: Boolean
Navigation Height	Specifies the height of the frame for holding the crosstab. Enter a numeric value to specify the height. If no value is given here, JReport will calculate the height. This property takes effect only when Enable Navigation is set to true. Data type: Float
Navigation Width	Specifies the width of the frame for holding the crosstab. Enter a numeric value to specify the width. If no value is given here, JReport will calculate the width. This property takes effect only when Enable Navigation is set to true. Data type: Float
Position	Specifies the position of the object. Available when the object is directly contained in the report body or a tabular cell. • **absolute** - The object's position will be decided by its X and Y property values. • **static** - The object will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled. Data type: Enumeration
Crosstab Property	
Column Totals on Left	Specifies whether to display the total columns on the left of the aggregates. Data type: Boolean
Row Totals on Top	Specifies whether to display the total rows on the top of the aggregates. Data type: Boolean
Suppress Column Header	Specifies whether to suppress the column header in view mode. Data type: Boolean
Suppress Row Header	Specifies whether to suppress the row header in view mode. Data type: Boolean

A crosstab has some child objects, the properties of which will be shown in the following topics:

- DBField
- Formula field
- Label

DBField

The properties of a DBField in a crosstab are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. This property is available only for aggregate field. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Foreground	Specifies the foreground color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Padding	
Bottom Padding	Specifies the space between the text and the bottom border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Left Padding	Specifies the space between the text and the left border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Right Padding	Specifies the space between the text and the right border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Top Padding	Specifies the space between the text and the top border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration
Text Format	
Auto Fit	Specifies whether to adjust the width of the object according to the contents. Data type: Boolean
Bold	Specifies whether to make the text bold. Data type: Boolean
Font Face	Specifies the font of the text. Choose an option from the drop-down list. Data type: Enumeration
Font Size	Specifies the font size of the text. Enter an integer value to change the size. Data type: Integer
Horizontal Alignment	Specifies the horizontal justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Italic	Specifies whether to make the text italic. Data type: Boolean
Strikethrough	Specifies whether to add a strikethrough line to the text. Data type: Boolean
Underline	Specifies whether to underline the text. Data type: Boolean
Vertical Alignment	Specifies the vertical justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Word Wrap	Specifies whether to wrap the text to the width. Data type: Boolean

•

Formula field

The properties of a formula field in a crosstab are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry (only for aggregate field)	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Foreground	Specifies the foreground color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Padding	
Bottom Padding	Specifies the space between the text and the bottom border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Left Padding	Specifies the space between the text and the left border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Right Padding	Specifies the space between the text and the right border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Top Padding	Specifies the space between the text and the top border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration
Text Format	
Auto Fit	Specifies whether to adjust the width of the object according to the contents. Data type: Boolean
Bold	Specifies whether to make the text bold. Data type: Boolean
Font Face	Specifies the font of the text. Choose an option from the drop-down list. Data type: Enumeration
Font Size	Specifies the font size of the text. Enter an integer value to change the size. Data type: Integer
Horizontal Alignment	Specifies the horizontal justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Italic	Specifies whether to make the text italic. Data type: Boolean
Strikethrough	Specifies whether to add a strikethrough line to the text. Data type: Boolean
Underline	Specifies whether to underline the text. Data type: Boolean
Vertical Alignment	Specifies the vertical justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Word Wrap	Specifies whether to wrap the text to the width. Data type: Boolean

Label

The header of a total row or column, the display name of a column or row field, and the display name of an aggregate field are represented as labels in the Report Structure panel of the Report Inspector. Their class types are CTTextField, CTDBTitleField, and CTLabelField respectively.

The properties of a label in a crosstab are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Text Format	
Auto Fit	Specifies whether to adjust the width of the object according to the contents. Data type: Boolean
Bold	Specifies whether to make the text bold. Data type: Boolean
Font Face	Specifies the font of the text. Choose an option from the drop-down list. Data type: Enumeration
Font Size	Specifies the font size of the text. Enter an integer value to change the size. Data type: Integer
Horizontal Alignment	Specifies the horizontal justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Italic	Specifies whether to make the text italic. Data type: Boolean
Strikethrough	Specifies whether to add a strikethrough line to the text. Data type: Boolean
Text	Specifies the text in the label. Enter a string to display as the label text. Data type: String
Underline	Specifies whether to underline the text. Data type: Boolean
Vertical Alignment	Specifies the vertical justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Word Wrap	Specifies whether to wrap the text to the width. Data type: Boolean
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Foreground	Specifies the foreground color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Padding	
Bottom Padding	Specifies the space between the text and the bottom border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Left Padding	Specifies the space between the text and the left border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Right Padding	Specifies the space between the text and the right border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Top Padding	Specifies the space between the text and the top border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration

DBField

The properties of a DBField in a web report are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. Enter a numeric value to change the position in inches. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. Enter a numeric value to change the position in inches. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Foreground	Specifies the foreground color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Padding	
Bottom Padding	Specifies the space between the text and the bottom border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Left Padding	Specifies the space between the text and the left border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Right Padding	Specifies the space between the text and the right border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Top Padding	Specifies the space between the text and the top border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration
Text Format	
Auto Fit	Specifies whether to adjust the width and height of the object according to the contents. Data type: Boolean
Bold	Specifies whether to make the text bold. Data type: Boolean
Font Face	Specifies the font of the text. Choose an option from the drop-down list. Data type: Enumeration
Font Size	Specifies the font size of the text. Enter an integer value to change the size. Data type: Integer
Horizontal Alignment	Specifies the horizontal justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Italic	Specifies whether to make the text italic. Data type: Boolean
Strikethrough	Specifies whether to add a strikethrough line to the text. Data type: Boolean
Underline	Specifies whether to underline the text. Data type: Boolean
Vertical Alignment	Specifies the vertical justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Word Wrap	Specifies whether to wrap the text to the width. Data type: Boolean
Others	
Export to Applet	Specifies whether to include the object when exporting the report to Applet. Data type: Boolean
Export to CSV	Specifies whether to include the object when exporting the report to CSV. Data type: Boolean
Export to Excel	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
Export to HTML	Specifies whether to include the object when exporting the report to HTML. Data type: Boolean
Export to PDF	Specifies whether to include the object when exporting the report to PDF. Data type: Boolean
Export to PostScript	Specifies whether to include the object when exporting the report to PostScript. Data type: Boolean
Export to Report Result	Specifies whether to include the object when exporting the report to Report Result. Data type: Boolean
Export to RTF	Specifies whether to include the object when exporting the report to RTF. Data type: Boolean
Export to Text	Specifies whether to include the object when exporting the report to Text. Data type: Boolean
Export to XML	Specifies whether to include the object when exporting the report to XML. Data type: Boolean

Filter control

The properties of a filter control are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Foreground	Specifies the foreground color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration
Title	
Background	Specifies the background color of the title bar. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Bold	Specifies whether to make the text bold in the title bar. Data type: Boolean
Font Face	Specifies the font of the text in the title bar. Choose an option from the drop-down list. Data type: Enumeration
Font Size	Specifies the font size of the text in the title bar. Enter an integer value to change the size. Data type: Integer
Foreground	Specifies the foreground color of the title bar. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Horizontal Alignment	Specifies the horizontal justification of the text in the title bar. Choose an option from the drop-down list. Data type: Enumeration
Italic	Specifies whether to make the text italic in the title bar. Data type: Boolean
Map Name	Specifies whether to use the name of the DBField added to the filter control as the text displayed in the title bar. Data type: Boolean
Show Title	Specifies whether to show the title bar. It is meaningful to set all the other properties in the Title category when this property is set to true. Data type: Boolean
Text	Specifies the text displayed in the title bar. This property can be edited only when Map Name is set to false. Enter a string to change the text. Data type: String
Underline	Specifies whether to underline the text in the title bar. Data type: Boolean
Others	
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. You can also use a formula to control whether or not to show the object. Data type: Boolean
Position	Specifies the position of the object. Choose an option from the drop-down list. • **static** - The object is positioned at the location in which it is inserted. If selected, the X, Y and other position-related properties will be hidden or disabled. The position of the object is affected by preceding objects in the same container. • **absolute** - The object will be located at the position specified by dragging and dropping or by setting its X and Y property values. The position of the object is not affected by preceding objects in the same container. Data type: Enumeration

Formula field

The properties of a formula field in a web report are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. Enter a numeric value to change the position in inches. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. Enter a numeric value to change the position in inches. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Foreground	Specifies the foreground color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Padding	
Bottom Padding	Specifies the space between the text and the bottom border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Left Padding	Specifies the space between the text and the left border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Right Padding	Specifies the space between the text and the right border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Top Padding	Specifies the space between the text and the top border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration
Text Format	
Auto Fit	Specifies whether to adjust the width and height of the object according to the contents. Data type: Boolean
Bold	Specifies whether to make the text bold. Data type: Boolean
Font Face	Specifies the font of the text. Choose an option from the drop-down list. Data type: Enumeration
Font Size	Specifies the font size of the text. Enter an integer value to change the size. Data type: Integer
Horizontal Alignment	Specifies the horizontal justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Italic	Specifies whether to make the text italic. Data type: Boolean
Strikethrough	Specifies whether to add a strikethrough line to the text. Data type: Boolean
Underline	Specifies whether to underline the text. Data type: Boolean
Vertical Alignment	Specifies the vertical justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Word Wrap	Specifies whether to wrap the text to the width. Data type: Boolean
Others	
Export to CSV	Specifies whether to include the object when exporting the report to CSV. Data type: Boolean
Export to Excel	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean

Image

The properties of an image in a web report are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Others	
Export to Applet	Specifies whether to include the object when exporting the report to Applet. Data type: Boolean
Export to CSV	Specifies whether to include the object when exporting the report to CSV. Data type: Boolean
Export to Excel	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
Export to HTML	Specifies whether to include the object when exporting the report to HTML. Data type: Boolean
Export to PDF	Specifies whether to include the object when exporting the report to PDF. Data type: Boolean
Export to PostScript	Specifies whether to include the object when exporting the report to PostScript. Data type: Boolean
Export to Report Result	Specifies whether to include the object when exporting the report to Report Result. Data type: Boolean
Export to RTF	Specifies whether to include the object when exporting the report to RTF. Data type: Boolean
Export to Text	Specifies whether to include the object when exporting the report to Text. Data type: Boolean
Export to XML	Specifies whether to include the object when exporting the report to XML. Data type: Boolean
Horizontal Alignment	Specifies the horizontal alignment of the image in the container. Choose an option from the drop-down list. Data type: Enumeration
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. You can also use a formula to control whether or not to show the object. Data type: Boolean
Position	Specifies the position of the object. Available when the object is directly contained in the report body or a tabular cell. • **absolute** - The object's position will be decided by its X and Y property values. • **static** - The object will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled. Data type: Enumeration
Vertical Alignment	Specifies the vertical alignment of the image in the container. Choose an option from the drop-down list. Data type: Enumeration
Image Property	
Alternate Text	Specifies the text that will be displayed instead if the image cannot be displayed. Data type: String
Disabled	Specifies whether to disable the actions on the image. Data type: Boolean
Image Name	Specifies the file name of the image. The image can be one on your local file system or retrieved by using a URL. Data type: String
Maximum Scaling Ratio	Specifies the maximum ratio up to which the image can be scaled. Data type: Float
Name	Specifies the name of the image. Data type: String
Rotation	Specifies the angle at which to rotate the image, in degrees. The following is the meaning of different values: • 0 - No rotation. • positive value - Rotates the image clockwise. • negative value - Rotates the image anticlockwise. Data type: Float
Scaling Mode	Specifies the scaling mode of the image, which controls the image behavior when you try to adjust the image field size. Choose an option from the drop-down list. • **actual size** - The image will remain its original size. If the image field size is smaller than the image size, part of the image will be cut off. • **fit image** - The image size will be automatically adjusted to fill up the image field, with its original perspective remained but under the limitation of Maximum Scaling Ratio. • **fit width** - The image will be automatically adjusted to fit the width of the image field but under the limitation of Maximum Scaling Ratio. • **fit height** - The image will be automatically adjusted to fit the height of the image field but under the limitation of Maximum Scaling Ratio. • **customize** - The image size will be automatically adjusted to fit the image field, regardless of the Maximum Scaling Ratio's value. Data type: Enumeration
Title	Specifies the tip information about the image, which will be displayed when you hover the mouse pointer over the image in Web Report result. Data type: String

Label

The properties of a label in a web report are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Text Format	
Auto Fit	Specifies whether to adjust the width and height of the object according to the contents. Data type: Boolean
Bold	Specifies whether to make the text bold. Data type: Boolean
Font Face	Specifies the font of the text. Choose an option from the drop-down list. Data type: Enumeration
Font Size	Specifies the font size of the text. Enter an integer value to change the size. Data type: Integer
Horizontal Alignment	Specifies the horizontal justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Italic	Specifies whether to make the text italic. Data type: Boolean
Strikethrough	Specifies whether to add a strikethrough line to the text. Data type: Boolean
Text	Specifies the text in the label. Enter a string to display as the label text. Data type: String
Underline	Specifies whether to underline the text. Data type: Boolean
Vertical Alignment	Specifies the vertical justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Word Wrap	Specifies whether to wrap the text to the width. Data type: Boolean
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Foreground	Specifies the foreground color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Others	
Export to Applet	Specifies whether to include the object when exporting the report to Applet. Data type: Boolean
Export to CSV	Specifies whether to include the object when exporting the report to CSV. Data type: Boolean
Export to Excel	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
Export to HTML	Specifies whether to include the object when exporting the report to HTML. Data type: Boolean
Export to PDF	Specifies whether to include the object when exporting the report to PDF. Data type: Boolean
Export to PostScript	Specifies whether to include the object when exporting the report to PostScript. Data type: Boolean
Export to Report Result	Specifies whether to include the object when exporting the report to Report Result. Data type: Boolean
Export to RTF	Specifies whether to include the object when exporting the report to RTF. Data type: Boolean
Export to Text	Specifies whether to include the object when exporting the report to Text. Data type: Boolean
Export to XML	Specifies whether to include the object when exporting the report to XML. Data type: Boolean
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. For an independent label, you can also use a formula to control whether or not to show it. Data type: Boolean
Position	Specifies the position of the object. Available when the object is directly contained in the report body or a tabular cell. • **absolute** - The object's position will be decided by its X and Y property values. • **static** - The object will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled. Data type: Enumeration
Padding	
Bottom Padding	Specifies the space between the text and the bottom border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Left Padding	Specifies the space between the text and the left border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Right Padding	Specifies the space between the text and the right border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Top Padding	Specifies the space between the text and the top border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration

Multimedia object (OLE object)

The properties of a multimedia object which is also called OLE object in a web report are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Others	
Export to Applet	Specifies whether to include the object when exporting the report to Applet. Data type: Boolean
Export to CSV	Specifies whether to include the object when exporting the report to CSV. Data type: Boolean
Export to Excel	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
Export to HTML	Specifies whether to include the object when exporting the report to HTML. Data type: Boolean
Export to PDF	Specifies whether to include the object when exporting the report to PDF. Data type: Boolean
Export to PostScript	Specifies whether to include the object when exporting the report to PostScript. Data type: Boolean
Export to Report Result	Specifies whether to include the object when exporting the report to Report Result. Data type: Boolean
Export to RTF	Specifies whether to include the object when exporting the report to RTF. Data type: Boolean
Export to Text	Specifies whether to include the object when exporting the report to Text. Data type: Boolean
Export to XML	Specifies whether to include the object when exporting the report to XML. Data type: Boolean
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. You can also use a formula to control whether or not to show the object. Data type: Boolean
Position	Specifies the position of the object. Available when the object is directly contained in the report body or a tabular cell. • **absolute** - The object's position will be decided by its X and Y property values. • **static** - The object will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled. Data type: Enumeration
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration
Object Settings	
Archive	Specifies the character string that can be used to implement your own archive functionality for the object. Data type: String
Class ID	Specifies the class identifier of the object. Data type: String
Code Base	Specifies the base path used to resolve relative URIs specified by the Class ID and Archive properties. When null, its default value is the base URI of the current report. Data type: String
Value Type	Specifies the MIME (Multipurpose Internet Mail Extensions) type of the object. Data type: String

Navigation control

The properties of a navigation control are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Others	
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. You can also use a formula to control whether or not to show the object. Data type: Boolean
Position	Specifies the position of the object. Choose an option from the drop-down list. • **static** - The object is positioned at the location in which it is inserted. If selected, the X, Y and other position-related properties will be hidden or disabled. The position of the object is affected by preceding objects in the same container. • **absolute** - The object will be located at the position specified by dragging and dropping or by setting its X and Y property values. The position of the object is not affected by preceding objects in the same container. Data type: Enumeration
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration

A navigation control has a child object:

- Navigation item

Navigation item

The properties of a navigation item are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Foreground	Specifies the foreground color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Text Format	
Auto Fit	Specifies whether to adjust the width and height of the object according to the contents. Data type: Boolean
Bold	Specifies whether to make the text bold. Data type: Boolean
Font Face	Specifies the font of the text. Choose an option from the drop-down list. Data type: Enumeration
Font Size	Specifies the font size of the text. Enter an integer value to change the size. Data type: Integer
Horizontal Alignment	Specifies the horizontal justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Italic	Specifies whether to make the text italic. Data type: Boolean
Strikethrough	Specifies whether to add a strikethrough line to the text. Data type: Boolean
Underline	Specifies whether to underline the text. Data type: Boolean
Vertical Alignment	Specifies the vertical justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Word Wrap	Specifies whether to wrap the text to the width. Data type: Boolean
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration
Padding	
Bottom Padding	Specifies the space between the text and the bottom border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Left Padding	Specifies the space between the text and the left border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Right Padding	Specifies the space between the text and the right border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Top Padding	Specifies the space between the text and the top border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Others	
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. You can also use a formula to control whether or not to show the object. Data type: Boolean
Position	Specifies the position of the object. Choose an option from the drop-down list. • **static** - The object is positioned at the location in which it is inserted. If selected, the X, Y and other position-related properties will be hidden or disabled. The position of the object is affected by preceding objects in the same container. • **absolute** - The object will be located at the position specified by dragging and dropping or by setting its X and Y property values. The position of the object is not affected by preceding objects in the same container. Data type: Enumeration

Page footer panel

The properties of a page footer panel in a web report are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Others	
Export to CSV	Specifies whether to include the object when exporting the report to CSV. Data type: Boolean
Export to Excel	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
Export to Report Result	Specifies whether to include the object when exporting the report to Report Result. Data type: Boolean
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. You can also use a formula to control whether or not to show the object. Data type: Boolean
Position	Specifies the position of the object. Available when the object is directly contained in the report body or a tabular cell. • **absolute** - The object's position will be decided by its X and Y property values. • **static** - The object will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled. Data type: Enumeration
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration

Page header panel

The properties of a page header panel in a web report are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Color	
Background	Specifies the background color and fill effect of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Others	
Export to CSV	Specifies whether to include the object when exporting the report to CSV. Data type: Boolean
Export to Excel	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
Export to Report Result	Specifies whether to include the object when exporting the report to Report Result. Data type: Boolean
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. You can also use a formula to control whether or not to show the object. Data type: Boolean
Position	Specifies the position of the object. Available when the object is directly contained in the report body or a tabular cell. • **absolute** - The object's position will be decided by its X and Y property values. • **static** - The object will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled. Data type: Enumeration
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration

Parameter control

The properties of a parameter control are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Foreground	Specifies the foreground color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Text Format	
Bold	Specifies whether to make the text bold. Data type: Boolean
Font Face	Specifies the font of the text. Choose an option from the drop-down list. Data type: Enumeration
Font Size	Specifies the font size of the text. Enter an integer value to change the size. Data type: Integer
Horizontal Alignment	Specifies the horizontal justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Italic	Specifies whether to make the text italic. Data type: Boolean
Underline	Specifies whether to underline the text. Data type: Boolean
Vertical Alignment	Specifies the vertical justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to specify the color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration
Padding	
Bottom Padding	Specifies the space between the text of the parameter control and its bottom border. Data type: Float
Left Padding	Specifies the space between the text of the parameter control and its left border. Data type: Float
Right Padding	Specifies the space between the text of the parameter control and its right border. Data type: Float
Top Padding	Specifies the space between the text of the parameter control and its top border. Data type: Float
Others	
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. You can also use a formula to control whether or not to show the object. Data type: Boolean
Position	Specifies the position of the object. Choose an option from the drop-down list. • **static** - The object is positioned at the location in which it is inserted. If selected, the X, Y and other position-related properties will be hidden or disabled. The position of the object is affected by preceding objects in the same container. • **absolute** - The object will be located at the position specified by dragging and dropping or by setting its X and Y property values. The position of the object is not affected by preceding objects in the same container. Data type: Enumeration

Parameter field

The properties of a parameter field in a web report are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Foreground	Specifies the foreground color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Padding	
Bottom Padding	Specifies the space between the text and the bottom border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Left Padding	Specifies the space between the text and the left border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Right Padding	Specifies the space between the text and the right border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Top Padding	Specifies the space between the text and the top border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration
Text Format	
Auto Fit	Specifies whether to adjust the width and height of the object according to the contents. Data type: Boolean
Bold	Specifies whether to make the text bold. Data type: Boolean
Font Face	Specifies the font of the text. Choose an option from the drop-down list. Data type: Enumeration
Font Size	Specifies the font size of the text. Enter an integer value to change the size. Data type: Integer
Horizontal Alignment	Specifies the horizontal justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Italic	Specifies whether to make the text italic. Data type: Boolean
Strikethrough	Specifies whether to add a strikethrough line to the text. Data type: Boolean
Underline	Specifies whether to underline the text. Data type: Boolean
Vertical Alignment	Specifies the vertical justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Word Wrap	Specifies whether to wrap the text to the width. Data type: Boolean
Others	
Export to CSV	Specifies whether to include the object when exporting the report to CSV. Data type: Boolean
Export to Excel	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
Export to Report Result	Specifies whether to include the object when exporting the report to Report Result. Data type: Boolean
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. For an independent parameter, you can also use a formula to control whether or not to show it. Data type: Boolean
Position	Specifies the position of the object. Available when the object is directly contained in the report body or a tabular cell. • **absolute** - The object's position will be decided by its X and Y property values. • **static** - The object will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled. Data type: Enumeration

Parameter form control

The properties of a parameter form control are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to specify the color. Data type: String
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to specify the color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration
Title	
Background	Specifies the background color of the title bar. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to specify the color. Data type: String
Bold	Specifies whether to make the text bold in the title bar. Data type: Boolean
Font Face	Specifies the font of the text in the title bar. Choose an option from the drop-down list. Data type: Enumeration
Font Size	Specifies the font size of the text in the title bar. Enter an integer value to change the size. Data type: Integer
Foreground	Specifies the foreground color of the title bar. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to specify the color. Data type: String
Horizontal Alignment	Specifies the horizontal justification of the text in the title bar. Choose an option from the drop-down list. Data type: Enumeration
Italic	Specifies whether to make the text italic in the title bar. Data type: Boolean
Show Title	Specifies whether to show the title bar. It is meaningful to set all the other properties in the Title category when this property is set to true. Data type: Boolean
Text	Specifies the text displayed in the title bar. Enter a string to change the text. Data type: String
Underline	Specifies whether to underline the text in the title bar. Data type: Boolean
Others	
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. You can also use a formula to control whether or not to show the object. Data type: Boolean
Position	Specifies the position of the object. Choose an option from the drop-down list. • **static** - The object is positioned at the location in which it is inserted. If selected, the X, Y and other position-related properties will be hidden or disabled. The position of the object is affected by preceding objects in the same container. • **absolute** - The object will be located at the position specified by dragging and dropping or by setting its X and Y property values. The position of the object is not affected by preceding objects in the same container. Data type: Enumeration

Report body

The properties of the report body in a web report are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Color	
Background	Specifies the background color of the report body. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border	
Border Color	Specifies the color of the borders. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the report body. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the report body. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the report body. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the report body. Choose a style from the drop-down list. Data type: Enumeration

Special field

The properties of a special field in a web report are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Foreground	Specifies the foreground color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Padding	
Bottom Padding	Specifies the space between the text and the bottom border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Left Padding	Specifies the space between the text and the left border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Right Padding	Specifies the space between the text and the right border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Top Padding	Specifies the space between the text and the top border of the object, in inches. Enter a numeric value to change the padding. Data type: Float
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration
Text Format	
Auto Fit	Specifies whether to adjust the width and height of the object according to the contents. Data type: Boolean
Bold	Specifies whether to make the text bold. Data type: Boolean
Font Face	Specifies the font of the text. Choose an option from the drop-down list. Data type: Enumeration
Font Size	Specifies the font size of the text. Enter an integer value to change the size. Data type: Integer
Horizontal Alignment	Specifies the horizontal justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Italic	Specifies whether to make the text italic. Data type: Boolean
Strikethrough	Specifies whether to add a strikethrough line to the text. Data type: Boolean
Underline	Specifies whether to underline the text. Data type: Boolean
Vertical Alignment	Specifies the vertical justification of the text in the object. Choose an option from the drop-down list. Data type: Enumeration
Word Wrap	Specifies whether to wrap the text to the width. Data type: Boolean
Others	
Export to Applet	Specifies whether to include the object when exporting the report to Applet. Data type: Boolean
Export to CSV	Specifies whether to include the object when exporting the report to CSV. Data type: Boolean
Export to Excel	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
Export to HTML	Specifies whether to include the object when exporting the report to HTML. Data type: Boolean
Export to PDF	Specifies whether to include the object when exporting the report to PDF. Data type: Boolean
Export to PostScript	Specifies whether to include the object when exporting the report to PostScript. Data type: Boolean
Export to Report Result	Specifies whether to include the object when exporting the report to Report Result. Data type: Boolean
Export to RTF	Specifies whether to include the object when exporting the report to RTF. Data type: Boolean
Export to Text	Specifies whether to include the object when exporting the report to Text. Data type: Boolean
Export to XML	Specifies whether to include the object when exporting the report to XML. Data type: Boolean
Field Type	Specifies another special field format to substitute the current special field. Choose an option from the drop-down list. Data type: Enumeration
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. For an independent special field, you can also use a formula to control whether or not to show it. Data type: Boolean
Position	Specifies the position of the object. Choose an option from the drop-down list. • **static** - The object is positioned at the location in which it is inserted. If selected, the X, Y and other position-related properties will be hidden or disabled. The position of the object is affected by preceding objects in the same container. • **absolute** - The object will be located at the position specified by dragging and dropping or by setting its X and Y property values. The position of the object is not affected by preceding objects in the same container. Data type: Enumeration **Note:** This property only affects objects with a parent container that is the report body, a table cell, or a tabular cell.

Table

The properties of a table in a web report are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
X	Specifies the horizontal coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Y	Specifies the vertical coordinate of the top left corner of the object, relative to its parent container. This property is unavailable if the Position property is set to static. Enter a numeric value to change the position in inches. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Others	
Enable Navigation	Specifies whether to enable page navigation on the table. If it is set to true, if the table contains more than one page, a page navigation bar specific for the table will be available right below it. You can use the bar to view the desired pages. When this property is set to true, you can further specify the height and width of the page for the table to display multiple pages, by setting the two properties Navigation Height and Navigation Width. If you later resize the table, the two properties will be reset to the resized values. Data type: Boolean
Export to Applet	Specifies whether to include the object when exporting the report to Applet. Data type: Boolean
Export to CSV	Specifies whether to include the object when exporting the report to CSV. Data type: Boolean
Export to Excel	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
Export to HTML	Specifies whether to include the object when exporting the report to HTML. Data type: Boolean
Export to PDF	Specifies whether to include the object when exporting the report to PDF. Data type: Boolean
Export to PostScript	Specifies whether to include the object when exporting the report to PostScript. Data type: Boolean
Export to Report Result	Specifies whether to include the object when exporting the report to Report Result. Data type: Boolean
Export to RTF	Specifies whether to include the object when exporting the report to RTF. Data type: Boolean
Export to Text	Specifies whether to include the object when exporting the report to Text. Data type: Boolean
Export to XML	Specifies whether to include the object when exporting the report to XML. Data type: Boolean
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. You can use a formula to control whether to show the object. Data type: Boolean
Navigation Height	Specifies the height of the page for the table to display multiple pages. Enter a numeric value to specify the height. If no value is given here, JReport will calculate the table page height. This property takes effect when Enable Navigation is set to true. Data type: Float
Navigation Width	Specifies the width of the page for the table to display multiple pages. Enter a numeric value to specify the width. If no value is given here, JReport will calculate the table page width. This property takes effect when Enable Navigation is set to true. Data type: Float
Position	Specifies the position of the object. Available when the object is directly contained in the report body or a tabular cell. • **absolute** - The object's position will be decided by its X and Y property values. • **static** - The object will be positioned at the default location in its container. If selected, the X, Y and other position-related properties will be hidden or disabled. Data type: Enumeration
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration

The following topics list properties of the children of a table:

- Table cell
- Table column
- Table detail row
- Table footer row
- Table group footer row
- Table group header row
- Table header row

Table cell

The properties of a table cell are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width.
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration
Others	
Export to Applet	Specifies whether to include the object when exporting the report to Applet. Data type: Boolean
Export to CSV	Specifies whether to include the object when exporting the report to CSV. Data type: Boolean
Export to Excel	Specifies whether to include the object when exporting the report to Excel. Data type: Boolean
Export to HTML	Specifies whether to include the object when exporting the report to HTML. Data type: Boolean
Export to PDF	Specifies whether to include the object when exporting the report to PDF. Data type: Boolean
Export to PostScript	Specifies whether to include the object when exporting the report to PostScript. Data type: Boolean
Export to Report Result	Specifies whether to include the object when exporting the report to Report Result. Data type: Boolean
Export to RTF	Specifies whether to include the object when exporting the report to RTF. Data type: Boolean
Export to Text	Specifies whether to include the object when exporting the report to Text. Data type: Boolean
Export to XML	Specifies whether to include the object when exporting the report to XML. Data type: Boolean

Table column

There are four types of columns in a table, which are common column, detail column, group column and summary column. These table column objects have the same properties as follows:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Others	
Auto Fit	Specifies whether to adjust the width of the object according to the contents. Data type: Boolean
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. Data type: Boolean

Table detail row

The properties of a table detail row are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Others	
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. Data type: Boolean
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration

Table footer row

The properties of a table footer row are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Others	
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. Data type: Boolean
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration

Table group footer row

The properties of a table group footer row are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Others	
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. Data type: Boolean
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration

Table group header row

The properties of a table group header row are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Others	
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. Data type: Boolean
Repeat	Specifies whether to make the group header appear in every page. Data type: Boolean
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration

Table header row

The properties of a table header row are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Others	
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. Data type: Boolean
Repeat on Pages	Specifies whether to make the object appear in every page of the report results. Data type: Boolean
Border	
Border Color	Specifies the color of the border of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Border Thickness	Specifies the width of the border in inches. Enter a numeric value to change the thickness. Data type: Float
Bottom Line	Specifies the line style of the bottom border of the object. Choose a style from the drop-down list. Data type: Enumeration
Left Line	Specifies the line style of the left border of the object. Choose a style from the drop-down list. Data type: Enumeration
Right Line	Specifies the line style of the right border of the object. Choose a style from the drop-down list. Data type: Enumeration
Top Line	Specifies the line style of the top border of the object. Choose a style from the drop-down list. Data type: Enumeration

Tabular

The properties of a tabular in a web report are:

Property Name	Description
Instance Name	Shows the instance name of the object. This property is read only.
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Others	
Invisible	Specifies whether to show or hide the object. All formulas and calculations will still be performed if the property is set to true. You can also use a formula to control whether or not to show the object. Data type: Boolean

A tabular has a child object:

- Tabular cell

Tabular cell

The properties of a tabular cell in a web report are:

Property Name	Description
General	
Display Name	Specifies the display name of the object. Data type: String
Instance Name	Shows the instance name of the object. This property is read only.
Geometry	
Height	Specifies the height of the object, in inches. Enter a numeric value to change the height. Data type: Float
Width	Specifies the width of the object, in inches. Enter a numeric value to change the width. Data type: Float
Color	
Background	Specifies the background color of the object. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xff0000), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String
Others	
Horizontal Alignment	Specifies the horizontal justification of the contents in the tabular cell. Choose an option from the drop-down list. Data type: Enumeration
Vertical Alignment	Specifies the vertical justification of the contents in the tabular cell. Choose an option from the drop-down list. Data type: Enumeration

Web report

The properties of a web report are:

Property Name	Description
Others	
Page Background	Specifies the background color of the report page. Choose a color from the drop-down list, enter a hexadecimal RGB value (for example, 0xCCAA00), or select Custom from the drop-down list to access the Color Picker dialog and specify a new color. Data type: String

Appendixes

This chapter presents additional information about JReport Server, including specific reference and several lists of properties, all of which will allow you to further understand JReport Server.

- Appendix 1: Properties in the server.properties file
- Appendix 2: Preference properties
- Appendix 3: System database properties in JReport Server Cluster
- Appendix 4: Mapping list of UI options and properties
- Appendix 5: Dynamic names
- Appendix 6: Parameters for running reports via URL
- Appendix 7: Properties for scheduling reports via URL
- Appendix 8: Page Report Studio tag library

Appendix 1: Properties in the server-properties File

The following table lists the properties that can be set in the server-properties file. Changes to properties will take effect after you restart the server unless you edit them in-game using a command.

Property Name	Default Value	Description
allow-flight	false	Determines if players can fly in survival mode if they have a mod that allows them to fly. If flying is allowed, monsters will appear in multiplayer worlds.
allow-nether	true	Determines if players can travel to the Nether.
broadcast-console-to-ops	true	Determines if console commands are broadcast to all online operators.
broadcast-rcon-to-ops	true	Determines if remote control commands are broadcast to all online operators.
difficulty	easy	Determines the difficulty level of the game. Valid values are peaceful, easy, normal, and hard.
enable-command-block	false	Determines if command blocks are enabled.
enable-query	false	Determines if the server responds to GameSpy4 protocol queries.
enable-rcon	false	Determines if remote control is enabled.
enable-status	true	Determines if the server responds to queries from the server list.
enforce-whitelist	false	Determines if only whitelisted players can join the server.
gamemode	0	Determines the default game mode for new players. Valid values are 0 (survival), 1 (creative), 2 (adventure), and 3 (spectator).
generator-settings		Determines the settings used in a flat world. Example: 3;minecraft:bedrock,2*minecraft:dirt,minecraft:grass_block;1;village
generate-structures	true	Determines if structures are generated in new chunks. Structures include villages, dungeons, and strongholds.
hardcore	false	Determines if the server is in hardcore mode. If hardcore is enabled, the difficulty is set to hard and players who die will be banned from the server.
level-name	world	Determines the name of the world to be loaded. Also used as the name of the folder in which the world is saved. Characters outside of the range of the ASCII character set may be used.
level-seed		Determines the seed used to generate the world. Example: -8372036854775808
level-type	default	Determines the type of world to be generated. Valid values are default, flat, and amplified.
max-chained-neighbor-updates	1000000	Determines the maximum number of consecutive block updates. Negative values remove the limit.
max-players	20	Determines the maximum number of players that can join the server at the same time.
max-tick-time	60000	Determines the maximum time in milliseconds that a single tick may take before the server is forcibly stopped. Negative values disable this feature.
max-world-size	29999984	Determines the maximum radius of the world in blocks. Valid values are 1 to 29999984.
network-compression-threshold	256	Determines the packet size in bytes beyond which packets are compressed. Negative values disable compression.
online-mode	true	Determines if the server authenticates players against Mojang's servers. Disabling this allows players with invalid usernames to join.
op-permission-level	4	Determines the default permission level for operators. Valid values are 1, 2, 3, and 4.
player-idle-timeout	0	Determines the time in minutes after which players will be kicked for inactivity. Zero disables this feature.
prevent-proxy-connections	false	Determines if players connecting from a proxy or VPN are prevented from joining the server.
pvp	true	Determines if players can attack other players.
query.port	25565	Determines the port used for GameSpy4 protocol queries. Valid values are 1 to 65535.
rcon.password		Determines the password required for remote control. Example: 7Sf6Ldq2
rcon.port	25575	Determines the port used for remote control. Valid values are 1 to 65535.
resource-pack		Determines the URL of a resource pack that players will be prompted to download when they join the server. Example: https://www.example.com/respack.zip
resource-pack-sha1		Determines the SHA1 checksum of the resource pack. Example: 2c3f0a1b4d5e6f7a8b9c0d1e2f3a4b5c6d7e8f9a
server-port	25565	Determines the port used for the server. Valid values are 1 to 65535.
server-ip		Determines the IP address to bind the server to. Example: 127.0.0.1
spawn-animals	true	Determines if passive mobs are spawned.
spawn-monsters	true	Determines if hostile mobs are spawned.
spawn-npcs	true	Determines if villagers are spawned.
spawn-protection	16	Determines the radius of the spawn protection area. Valid values are 0 to 29999984.
sync-chunk-writes	true	Determines if chunk writes are synchronized. Disabling this may improve performance but can cause data loss.
text-filtering-config		Determines the configuration for text filtering. Example: 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,118,119,120,121,122,123,124,125,126,127,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,144,145,146,147,148,149,150,151,152,153,154,155,156,157,158,159,160,161,162,163,164,165,166,167,168,169,170,171,172,173,174,175,176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,191,192,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,208,209,210,211,212,213,214,215,216,217,218,219,220,221,222,223,224,225,226,227,228,229,230,231,232,233,234,235,236,237,238,239,240,241,242,243,244,245,246,247,248,249,250,251,252,253,254,255,256,257,258,259,260,261,262,263,264,265,266,267,268,269,270,271,272,273,274,275,276,277,278,279,280,281,282,283,284,285,286,287,288,289,290,291,292,293,294,295,296,297,298,299,300,301,302,303,304,305,306,307,308,309,310,311,312,313,314,315,316,317,318,319,320,321,322,323,324,325,326,327,328,329,330,331,332,333,334,335,336,337,338,339,340,341,342,343,344,345,346,347,348,349,350,351,352,353,354,355,356,357,358,359,360,361,362,363,364,365,366,367,368,369,370,371,372,373,374,375,376,377,378,379,380,381,382,383,384,385,386,387,388,389,390,391,392,393,394,395,396,397,398,399,400,401,402,403,404,405,406,407,408,409,410,411,412,413,414,415,416,417,418,419,420,421,422,423,424,425,426,427,428,429,430,431,432,433,434,435,436,437,438,439,440,441,442,443,444,445,446,447,448,449,450,451,452,453,454,455,456,457,458,459,460,461,462,463,464,465,466,467,468,469,470,471,472,473,474,475,476,477,478,479,480,481,482,483,484,485,486,487,488,489,490,491,492,493,494,495,496,497,498,499,500,501,502,503,504,505,506,507,508,509,510,511,512,513,514,515,516,517,518,519,520,521,522,523,524,525,526,527,528,529,530,531,532,533,534,535,536,537,538,539,540,541,542,543,544,545,546,547,548,549,550,551,552,553,554,555,556,557,558,559,560,561,562,563,564,565,566,567,568,569,570,571,572,573,574,575,576,577,578,579,580,581,582,583,584,585,586,587,588,589,590,591,592,593,594,595,596,597,598,599,600,601,602,603,604,605,606,607,608,609,610,611,612,613,614,615,616,617,618,619,620,621,622,623,624,625,626,627,628,629,630,631,632,633,634,635,636,637,638,639,640,641,642,643,644,645,646,647,648,649,650,651,652,653,654,655,656,657,658,659,660,661,662,663,664,665,666,667,668,669,670,671,672,673,674,675,676,677,678,679,680,681,682,683,684,685,686,687,688,689,690,691,692,693,694,695,696,697,698,699,700,701,702,703,704,705,706,707,708,709,710,711,712,713,714,715,716,717,718,719,720,721,722,723,724,725,726,727,728,729,730,731,732,733,734,735,736,737,738,739,740,741,742,743,744,745,746,747,748,749,750,751,752,753,754,755,756,757,758,759,760,761,762,763,764,765,766,767,768,769,770,771,772,773,774,775,776,777,778,779,780,781,782,783,784,785,786,787,788,789,790,791,792,793,794,795,796,797,798,799,800,801,802,803,804,805,806,807,808,809,810,811,812,813,814,815,816,817,818,819,820,821,822,823,824,825,826,827,828,829,830,831,832,833,834,835,836,837,838,839,840,841,842,843,844,845,846,847,848,849,850,851,852,853,854,855,856,857,858,859,860,861,862,863,864,865,866,867,868,869,870,871,872,873,874,875,876,877,878,879,880,881,882,883,884,885,886,887,888,889,890,891,892,893,894,895,896,897,898,899,900,901,902,903,904,905,906,907,908,909,910,911,912,913,914,915,916,917,918,919,920,921,922,923,924,925,926,927,928,929,930,931,932,933,934,935,936,937,938,939,940,941,942,943,944,945,946,947,948,949,950,951,952,953,954,955,956,957,958,959,960,961,962,963,964,965,966,967,968,969,970,971,972,973,974,975,976,977,978,979,980,981,982,983,984,985,986,987,988,989,990,991,992,993,994,995,996,997,998,999,1000,1001,1002,1003,1004,1005,1006,1007,1008,1009,1010,1011,1012,1013,1014,1015,1016,1017,1018,1019,1020,1021,1022,1023,1024,1025,1026,1027,1028,1029,1030,1031,1032,1033,1034,1035,1036,1037,1038,1039,1040,1041,1042,1043,1044,1045,1046,1047,1048,1049,1050,1051,1052,1053,1054,1055,1056,1057,1058,1059,1060,1061,1062,1063,1064,1065,1066,1067,1068,1069,1070,1071,1072,1073,1074,1075,1076,1077,1078,1079,1080,1081,1082,1083,1084,1085,1086,1087,1088,1089,1090,1091,1092,1093,1094,1095,1096,1097,1098,1099,1100,1101,1102,1103,1104,1105,1106,1107,1108,1109,1110,1111,1112,1113,1114,1115,1116,1117,1118,1119,1120,1121,1122,1123,1124,1125,1126,1127,1128,1129,1130,1131,1132,1133,1134,1135,1136,1137,1138,1139,1140,1141,1142,1143,1144,1145,1146,1147,1148,1149,1150,1151,1152,1153,1154,1155,1156,1157,1158,1159,1160,1161,1162,1163,1164,1165,1166,1167,1168,1169,1170,1171,1172,1173,1174,1175,1176,1177,1178,1179,1180,1181,1182,1183,1184,1185,1186,1187,1188,1189,1190,1191,1192,1193,1194,1195,1196,1197,1198,1199,1200,1201,1202,1203,1204,1205,1206,1207,1208,1209,1210,1211,1212,1213,1214,1215,1216,1217,1218,1219,1220,1221,1222,1223,1224,1225,1226,1227,1228,1229,1230,1231,1232,1233,1234,1235,1236,1237,1238,1239,1240,1241,1242,1243,1244,1245,1246,1247,1248,1249,1250,1251,1252,1253,1254,1255,1256,1257,1258,1259,1260,1261,1262,1263,1264,1265,1266,1267,1268,1269,1270,1271,1272,1273,1274,1275,1276,1277,1278,1279,1280,1281,1282,1283,1284,1285,1286,1287,1288,1289,1290,1291,1292,1293,1294,1295,1296,1297,1298,1299,1300,1301,1302,1303,1304,1305,1306,1307,1308,1309,1310,1311,1312,1313,1314,1315,1316,1317,1318,1319,1320,1321,1322,1323,1324,1325,1326,1327,1328,1329,1330,1331,1332,1333,1334,1335,1336,1337,1338,1339,1340,1341,1342,1343,1344,1345,1346,1347,1348,1349,1350,1351,1352,1353,1354,1355,1356,1357,1358,1359,1360,1361,1362,1363,1364,1365,1366,1367,1368,1369,1370,1371,1372,1373,1374,1375,1376,1377,1378,1379,1380,1381,1382,1383,1384,1385,1386,1387,1388,1389,1390,1391,1392,1393,1394,1395,1396,1397,1398,1399,1400,1401,1402,1403,1404,1405,1406,1407,1408,1409,1410,1411,1412,1413,1414,1415,1416,1417,1418,1419,1420,1421,1422,1423,1424,1425,1426,1427,1428,1429,1430,1431,1432,1433,1434,1435,1436,1437,1438,1439,1440,1441,1442,1443,1444,1445,1446,1447,1448,1449,1450,1451,1452,1453,1454,1455,1456,1457,1458,1459,1460,1461,1462,1463,1464,1465,1466,1467,1468,1469,1470,1471,1472,1473,1474,1475,1476,1477,1478,1479,1480,1481,1482,1483,1484,1485,1486,1487,1488,1489,1490,1491,1492,1493,1494,1495,1496,1497,1498,1499,1500,1501,1502,1503,1504,1505,1506,1507,1508,1509,1510,1511,1512,1513,1514,1515,1516,1517,1518,1519,1520,1521,1522,1523,1524,1525,1526,1527,1528,1529,1530,1531,1532,1533,1534,1535,1536,1537,1538,1539,1540,1541,1542,1543,1544,1545,1546,1547,1548,1549,1550,1551,1552,1553,1554,1555,1556,1557,1558,1559,1560,1561,1562,1563,1564,1565,1566,1567,1568,1569,1570,1571,1572,1573,1574,1575,1576,1577,1578,1579,1580,1581,1582,1583,1584,1585,1586,1587,1588,1589,1590,1591,1592,1593,1594,1595,1596,1597,1598,1599,1600,1601,1602,1603,1604,1605,1606,1607,1608,1609,1610,1611,1612,1613,1614,1615,1616,1617,1618,1619,1620,1621,1622,1623,1624,1625,1626,1627,1628,1629,1630,1631,1632,1633,1634,1635,1636,1637,1638,1639,1640,1641,1642,1643,1644,1645,1646,1647,1648,1649,1650,1651,1652,1653,1654,1655,1656,1657,1658,1659,1660,1661,1662,1663,1664,1665,1666,1667,1668,1669,1670,1671,1672,1673,1674,1675,1676,1677,1678,1679,1680,1681,1682,1683,1684,1685,1686,1687,1688,1689,1690,1691,1692,1693,1694,1695,1696,1697,1698,1699,1700,1701,1702,1703,1704,1705,1706,1707,1708,1709,1710,1711,1712,1713,1714,1715,1716,1717,1718,1719,1720,1721,1722,1723,1724,1725,1726,1727,1728,1729,1730,1731,1732,1733,1734,1735,1736,1737,1738,1739,1740,1741,1742,1743,1744,1745,1746,1747,1748,1749,1750,1751,1752,1753,1754,1755,1756,1757,1758,1759,1760,1761,1762,1763,1764,1765,1766,1767,1768,1769,1770,1771,1772,1773,1774,1775,1776,1777,1778,1779,1780,1781,1782,1783,1784,1785,1786,1787,1788,1789,1790,1791,1792,1793,1794,1795,1796,1797,1798,1799,1800,1801,1802,1803,1804,1805,1806,1807,1808,1809,1810,1811,1812,1813,1814,1815,1816,1817,1818,1819,1820,1821,1822,1823,1824,1825,1826,1827,1828,1829,1830,1831,1832,1833,1834,1835,1836,1837,1838,1839,1840,1841,1842,1843,1844,1845,1846,1847,1848,1849,1850,1851,1852,1853,1854,1855,1856,1857,1858,1859,1860,1861,1862,1863,1864,1865,1866,1867,1868,1869,1870,1871,1872,1873,1874,1875,1876,1877,1878,1879,1880,1881,1882,1883,1884,1885,1886,1887,1888,1889,1890,1891,1892,1893,1894,1895,1896,1897,1898,1899,1900,1901,1902,1903,1904,1905,1906,1907,1908,1909,1910,1911,1912,1913,1914,1915,1916,1917,1918,1919,1920,1921,1922,1923,1924,1925,1926,1927,1928,1929,1930,1931,1932,1933,1934,1935,1936,1937,1938,1939,1940,1941,1942,1943,1944,1945,1946,1947,1948,1949,1950,1951,1952,1953,1954,1955,1956,1957,1958,1959,1960,1961,1962,1963,1964,1965,1966,1967,1968,1969,1970,1971,1972,1973,1974,1975,1976,1977,1978,1979,1980,1981,1982,1983,1984,1985,1986,1987,1988,1989,1990,1991,1992,1993,1994,1995,1996,1997,1998,1999,2000,2001,2002,2003,2004,2005,2006,2007,2008,2009,2010,2011,2012,2013,2014,2015,2016,2017,2018,2019,2020,2021,2022,2023,2024,2025,2026,2027,2028,2029,2030,2031,2032,2033,2034,2035,2036,2037,2038,2039,2040,2041,2042,2043,2044,2045,2046,2047,2048,2049,2050,2051,2052,2053,2054,2055,2056,2057,2058,2059,2060,2061,2062,2063,2064,2065,2066,2067,2068,2069,2070,2071,2072,2073,2074,2075,2076,2077,2078,2079,2080,2081,2082,2083,2084,2085,2086,2087,2088,2089,2090,2091,2092,2093,2094,2095,2096,2097,2098,2099,2100,2101,2102,2103,2104,2105,2106,2107,2108,2109,2110,2111,2112,2113,2114,2115,2116,2117,2118,2119,2120,2121,2122,2123,2124,2125,2126,2127,2128,2129,2130,2131,2132,2133,2134,2135,2136,2137,2138,2139,2140,2141,2142,2143,2144,2145,2146,2147,2148,2149,2150,2151,2152,2153,2154,2155,2156,2157,2158,2159,2160,2161,2162,2163,2164,2165,2166,2167,2168,2169,2170,2171,2172,2173,2174,2175,2176,2177,2178,2179,2180,2181,2182,2183,2184,2185,2186,2187,2188,2189,2190,2191,2192,2193,2194,2195,2196,2197,2198,2199,2200,2201,2202,2203,2204,2205,2206,2207,2208,2209,2210,2211,2212,2213,2214,2215,2216,2217,2218,2219,2220,2221,2222,2223,2224,2225,2226,2227,2228,2229,2230,2231,2232,2233,2234,2235,2236,2237,2238,2239,2240,2241,2242,2243,2244,2245,2246,2247,2248,2249,2250,2251,2252,2253,2254,2255,2256,2257,2258,2259,2260,2261,2262,2263,2264,2265,2266,2267,2268,2269,2270,2271,2272,2273,2274,2275,2276,2277,2278,2279,2280,2281,2282,2283,2284,2285,2286,2287,2288,2289,2290,2291,2292,2293,2294,2295,2296,2297,2298,2299,2300,2301,2302,2303,2304,2305,2306,2307,2308,2309,2310,2311,2312,2313,2314,2315,2316,2317,2318,2319,2320,2321,2322,2323,2324,2325,2326,2327,2328,2329,2330,2331,2332,2333,2334,2335,2336,2337,2338,2339,2340,2341,2342,2343,2344,2345,2346,2347,2348,2349,2350,2351,2352,2353,2354,2355,2356,2357,2358,2359,2360,2361,2362,2363,2364,2365,2366,2367,2368,2369,2370,2371,2372,2373,2374,2375,2376,2377,2378,2379,2380,2381,2382,2383,2384,2385,2386,2387,2388,2389,2390,2391,2392,2393,2394,2395,2396,2397,2398,2399,2400,2401,2402,2403,2404,2405,2406,2407,2408,2409,2410,2411,2412,2413,2414,2415,2416,2417,2418,2419,2420,2421,2422,2423,2424,2425,2426,2427,2428,2429,2430,2431,2432,2433,2434,2435,2436,2437,2438,2439,2440,2441,2442,2443,2444,2445,2446,2447,2448,2449,2450,2451,2452,2453,2454,2455,2456,2457,2458,2459,2460,2461,2462,2463,2464,2465,2466,2467,2468,2469,2470,2471,2472,2473,2474,2475,2476,2477,2478,2479,2480,2481,2482,2483,2484,2485,2486,2487,2488,2489,2490,2491,2492,2493,2494,2495,2496,2497,2498,2499,2500,2501,2502,2503,2504,2505,2506,2507,2508,2509,2510,2511,2512,2513,2514,2515,2516,2517,2518,2519,2520,2521,2522,2523,2524,2525,2526,2527,2528,2529,2530,2531,2532,2533,2534,2535,2536,2537,2538,2539,2540,2541,2542,2543,2544,2545,2546,2547,2548,2549,2550,2551,2552,2553,2554,2555,2556,2557,2558,2559,2560,2561,2562,2563,2564,2565,2566,2567,2568,2569,2570,2571,2572,2573,2574,2575,2576,2577,2578,2579,2580,2581,2582,2583,2584,2585,2586,2587,2588,2589,2590,2591,2592,2593,2594,2595,2596,2597,2598,2599,2600,2601,2602,2603,2604,2605,2606,2607,2608,2609,2610,2611,2612,2613,2614,2615,2616,2617,2618,2619,2620,2621,2622,2623,2624,2625,2626,2627,2628,2629,2630,2631,2632,2633,2634,2635,2636,2637,2638,2639,2640,2641,2642,2643,2644,2645,2646,2647,2648,2649,2650,2651,2652,2653,2654,2655,2656,2657,2658,2659,2660,2661,2662,2663,2664,2665,2666,2667,2668,2669,2670,2671,2672,2673,2674,2675,2676,2677,2678,2679,2680,2681,2682,2683,2684,2685,2686,2687,2688,2689,2690,2691,2692,2693,2694,2695,2696,2697,2698,2699,2700,2701,2702,2703,2704,2705,2706,2707,2708,2709,2710,2711,2712,2713,2714,2715,2716,2717,2718,2719,2720,2721,2722,2723,2724,2725,2726,2727,2728,2729,2730,2731,2732,2733,2734,2735,2736,2737,2738,2739,2740,2741,2742,2743,2744,2745,2746,2747,2748,2749,2750,2751,2752,2753,2754,2755,2756,2757,2758,2759,2760,2761,2762,2763,2764,2765,2766,2767,2768,2769,2770,2771,2772,2773,2774,2775,2776,2777,2778,2779,2780,2781,2782,2783,2784,2785,2786,2787,2788,2789,2790,2791,2792,2793,2794,2795,2796,2797,2798,2799,2800,2801,2802,2803,2804,2805,2806,2807,2808,2809,2810,2811,2812,2813,2814,2815,2816,2817,2818,2819,2820,2821,2822,2823,2824,2825,2826,2827,2828,2829,2830,2831,2832,2833,2834,2835,2836,2837,2838,2839,2840,2841,2842,2843,2844,2845,2846,2847,2848,2849,2850,2851,2852,2853,2854,2855,2856,2857,2858,2859,2860,2861,2862,2863,2864,2865,2866,2867,2868,2869,2870,2871,2872,2873,2874,2875,2876,2877,2878,2879,2880,2881,2882,2883,2884,2885,2886,2887,2888,2889,2890,2891,2892,2893,2894,2895,2896,2897,2898,2899,2900,2901,2902,2903,2904,2905,2906,2907,2908,2909,2910,2911,2912,2913,2914,2915,2916,2917,2918,2919,2920,2921,2922,2923,2924,2925,2926,2927,2928,2929,2930,2931,2932,2933,2934,2935,2936,2937,2938,2939,2940,2941,2942,2943,2944,2945,2946,2947,2948,2949,2950,2951,2952,2953,2954,2955,2956,2957,2958,2959,2960,2961,2962,2963,2964,2965,2966,2967,2968,2969,2970,2971,2972,2973,2974,2975,2976,2977,2978,2979,2980,2981,2982,2983,2984,2985,2986,2987,2988,2989,2990,2991,2992,2993,2994,2995,2996,2997,2998,2999,3000,3001,3002,3003,3004,3005,3006,3007,3008,3009,3010,3011,3012,3013,3014,3015,3016,3017,3018,3019,3020,3021,3022,3023,3024,3025,3026,3027,3028,3029,3030,3031,3032,3033,3034,3035,3036,3037,3038,3039,3040,3041,3042,3043,3044,3045,3046,3047,3048,3049,3050,3051,3052,3053,3054,3055,3056,3057,3058,3059,3060,3061,3062,3063,3064,3065,3066,3067,3068,3069,3070,3071,3072,3073,3074,3075,3076,3077,3078,3079,3080,3081,3082,3083,3084,3085,3086,3087,3088,3089,3090,3091,3092,3093,3094,3095,3096,3097,3098,3099,3100,3101,3102,3103,3104,3105,3106,3107,3108,3109,3110,3111,3112,3113,3114,3115,3116,3117,3118,3119,3120,3121,3122,3123,3124,3125,3126,3127,3128,3129,3130,3131,3132,3133,3134,3135,3136,3137,3138,3139,3140,3141,3142,3143,3144,3145,3146,3147,3148,3149,3150,3151,3152,3153,3154,3155,3156,3157,3158,3159,3160,3161,3162,3163,3164,3165,3166,3167,3168,3169,3170,3171,3172,3173,3174,3175,3176,3177,3178,3179,3180,3181,3182,3183,3184,3185,3186,3187,3188,3189,3190,3191,3192,3193,3194,3195,3196,3197,3198,3199,3200,3201,3202,3203,3204,3205,3206,3207,3208,3209,3210,3211,3212,3213,3214,3215,3216,3217,3218,3219,3220,3221,3222,3223,3224,3225,3226,3227,3228,3229,3230,3231,3232,3233,3234,3235,3236,3237,3238,3239,3240,3241,3242,3243,3244,3245,3246,3247,3248,3249,3250,3251,3252,3253,3254,3255,3256,3257,3258,3259,3260,3261,3262,3263,3264,3265,3266,3267,3268,3269,3270,3271,3272,3273,3274,3275,3276,3277,3278,3279,3280,3281,3282,3283,3284,3285,3286,3287,3288,3289,3290,3291,3292,3293,3294,3295,3296,3297,3298,3299,3300,3301,3302,3303,3304,3305,3306,3307,3308,3309,3310,3311,3312,3313,3314,3315,3316,3317,3318,3319,3320,3321,3322,3323,3324,3325,3326,3327,3328,3329,3330,3331,3332,3333,3334,3335,3336,3337,3338,3339,3340,3341,3342,3343,3344,3345,3346,3347,3348,3349,3350,3351,3352,3353,3354,3355,3356,3357,3358,3359,3360,3361,3362,3363,3364,3365,3366,3367,3368,3369,3370,3371,3372,3373,3374,3375,3376,3377,3378,3379,3380,3381,3382,3383,3384,3385,3386,3387,3388,3389,3390,3391,3392,3393,3394,3395,3396,3397,3398,3399,3400,3401,3402,3403,3404,3405,3406,3407,3408,3409,3410,3411,3412,3413,3414,3415,3416,3417,3418,3419,3420,3421,3422,3423,3424,3425,3426,3427,3428,3429,3430,3431,3432,3433,3434,3435,3436,3437,3438,3439,3440,3441,3442,3443,3444,3445,3446,3447,3448,3449,3450,3451,3452,3453,3454,3455,3456,3457,3458,3459,3460,3461,3462,3463,3464,3465,3466,3467,3468,3469,3470,3471,3472,3473,3474,3475,3476,3477,3478,3479,3480,3481,3482,3483,3484,3485,3486,3487,3488,3489,3490,3491,3492,3493,3494,3495,3496,3497,3498,3499,3500,3501,3502,3503,3504,3505,3506,3507,3508,3509,3510,3511,3512,3513,3514,3515,3516,3517,3518,3519,3520,3521,3522,3523,3524,3525,3526,3527,3528,3529,3530,3531,3532,3533,3534,3535,3536,3537,3538,3539,3540,3541,3542,3543,3544,3545,3546,3547,3548,3549,3550,3551,3552,3553,3554,3555,3556,3557,3558,3559,3560,3561,3562,3563,3564,3565,3566,3567,3568,3569,3570,3571,3572,3573,3574,3575,3576,3577,3578,3579,3580,3581,3582,3583,3584,3585,3586,3587,3588,3589,3590,3591,3592,3593,3594,3595,3596,3597,3598,3599,3600,3601,3602,3603,3604,3605,3606,3607,3608,3609,3610,3611,3612,3613,3614,3615,3616,3617,3618,3619,3620,3621,3622,3623,3624,3625,3626,3627,3628,3629,3630,3631,3632,3633,3634,3635,3636,3637,3638,3639,3640,3641,3642,3643,3644,3645,3646,3647,3648,3649,3650,3651,3652,3653,3654,3655,3656,3657,3658,3659,3660,3661,3662,3663,3664,3665,3666,3667,3668,3669,3670,3671,3672,3673,3674,3675,3676,3677,3678,3679,3680,3681,3682,3683,3684,3685,3686,3687,3688,3689,3690,3691,3692,3693,3694,3695,3696,3697,3698,3699,3700,3701,3702,3703,3704,3705,3706,3707,3708,3709,3710,3711,3712,3713,3714,3715,3716,3717,3718,3719,3720,3721,3722,3723,3724,3725,3726,3727,3728,3729,3730,3731,3732,3733,3734,3735,3736,3737,3738,3739,3740,3741,3742,3743,3744,3745,3746,3747,3748,3749,3750,3751,3752,3753,3754,3755,3756,3757,3758,3759,3760,3761,3762,3763,3764,3765,3766,3767,3768,3769,3770,3771,3772,3773,3774,3775,3776,3777,3778,3779,3780,3781,3782,3783,3784,3785,3786,3787,3788,3789,3790,3791,3792,3793,3794,3795,3796,3797,3798,3799,3800,3801,3802,3803,3804,3805,3806,3807,3808,3809,3810,3811,3812,3813,3814,3815,3816,3817,3818,3819,3820,3821,3822,3823,3824,3825,3826,3827,3828,3829,3830,3831,3832,3833,3834,3835,3836,3837,3838,3839,3840,3841,3842,3843,3844,3845,3846,3847,3848,3849,3850,3851,3852,3853,3854,3855,3856,3857,3858,3859,3860,3861,3862,3863,3864,3865,3866,3867,3868,3869,3870,3871,3872,3873,3874,3875,3876,3877,3878,3879,3880,3881,3882,3883,3884,3885,3886,3887,3888,3889,3890,3891,3892,3893,3894,3895,3896,3897,3898,3899,3900,3901,3902,3903,3904,3905,3906,3907,3908,3909,3910,3911,3912,3913,3914,3915,3916,3917,3918,3919,3920,3921,3922,3923,3924,3925,3926,3927,3928,3929,3930,3931,3932,3933,3934,3935,3936,3937,3938,3939,3940,3941,3942,3943,3944,3945,3946,3947,3948,3949,3950,3951,3952,3953,3954,3955,3956,3957,3958,3959,3960,3961,3962,3963,3964,3965,3966,3967,3968,3969,3970,3971,3972,3973,3974,3975,3976,3977,3978,3979,3980,3981,3982,3983,3984,3985,3986,3987,3988,3989,3990,3991,3992,3993,3994,3995,3996,3997,3998,3999,4000,4001,4002,4003,4004,4005,4006,4007,4008,4009,4010,4011,4012,4013,4014,4015,4016,4017,4018,4019,4020,4021,4022,4023,4024,4025,4026,4027,4028,4029,4030,4031,4032,4033,4034,4035,4036,4037,4038,4039,4040,4041,4042,4043,4044,4045,4046,4047,4048,4049,4050,4051,4052,4053,4054,4055,4056,4057,4058,4059,4060,4061,4062,4063,4064,4065,4066,4067,4068,4069,4070,4071,4072,4073,4074,4075,4076,4077,4078,4079,4080,4081,4082,4083,4084,4085,4086,4087,4088,4089,4090,4091,4092,4093,4094,4095,4096,4097,4098,4099,4100,4101,4102,4103,4104,4105,4106,4107,4108,4109,4110,4111,4112,4113,4114,4115,4116,4117,4118,4119,4120,4121,4122,4123,4124,4125,4126,4127,4128,4129,4130,4131,4132,4133,4134,4135,4136,4137,4138,4139,4140,4141,4142,4143,4144,4145,4146,4147,4148,4149,4150,4151,4152,4153,4154,4155,4156,4157,4158,4159,4160,4161,4162,4163,4164,4165,4166,4167,4168,4169,4170,4171,4172,4173,4174,4175,4176,4177,4178,4179,4180,4181,4182,4183,4184,4185,4186,4187,4188,4189,4190,4191,4192,4193,4194,4195,4196,4197,4198,4199,4200,4201,4202,4203,4204,4205,4206,4207,4208,4209,4210,4211,4212,4213,4214,4215,4216,4217,4218,4219,4220,4221,4222,4223,4224,4225,4226,4227,4228,4229,4230,4231,4232,4233,4234,4235,4236,4237,4238,4239,4240,4241,4242,4243,4244,4245,4246,4247,4248,4249,4250,4251,4252,4253,4254,4255,4256,4257,4258,4259,4260,4261,4262,4263,4264,4265,4266,4267,4268,4269,4270,4271,4272,4273,4274,4275,4276,4277,4278,4279,4280,4281,4282,4283,4284,4285,4286,4287,4288,4289,4290,4291,4292,4293,4294,4295,4296,4297,4298,4299,4300,4301,4302,4303,4304,4305,4306,4307,4308,4309,4310,4311,4312,4313,4314,4315,4316,4317,4318,4319,4320,4321,4322,4323,4324,4325,4326,4327,4328,4329,4330,4331,4332,4333,4334,4335,4336,4337,4338,4339,4340,4341,4342,4343,4344,4345,4346,4347,4348,4349,4350,4351,4352,4353,4354,4355,4356,4357,4358,4359,4360,4361,4362,4363,4364,4365,4366,4367,4368,4369,4370,4371,4372,4373,4374,4375,4376,4377,4378,4379,4380,4381,4382,4383,4384,4385,4386,4387,4388,4389,4390,4391,4392,4393,4394,4395,4396,4397,4398,4399,4400,4401,4402,4403,4404,4405,4406,4407,4408,4409,4410,4411,4412,4413,4414,4415,4416,4417,4418,4419,4420,4421,4422,4423,4424,4425,4426,4427,4428,4429,4430,4431,4432,4433,4434,4435,4436,4437,4438,4439,4440,4441,4442,4443,4444,4445,4446,4447,4448,4449,4450,4451,4452,4453,4454,4455,4456,4457,4458,4459,4460,4461,4462,4463,4464,4465,4466,4467,4468,4469,4470,4471,4472,4473,4474,4475,4476,4477,4478,4479,4480,4481,4482,4483,4484,4485,4486,4487,4488,4489,4490,4491,4492,4493,4494,4495,4496,4497,4498,4499,4500,4501,4502,4503,4504,4505,4506,4507,4508,4509,4510,4511,4512,4513,4514,4515,4516,4517,4518,4519,4520,4521,4522,4523,4524,4525,4526,4527,4528,4529,4530,4531,4532,4533,4534,4535,4536,4537,4538,4539,4540,4541,4542,4543,4544,4545,4546,4547,4548,4549,4550,4551,4552,4553,4554,4555,4556,4557,4558,4559,4560,4561,4562,4563,4564,4565,4566,4567,4568,4569,4570,4571,4572,4573,4574,4575,4576,4577,4578,4579,4580,4581,4582,4583,4584,4585,4586,4587,4588,4589,4590,4591,4592,4593,4594,4595,4596,4597,4598,4599,4600,4601,4602,4603,4604,4605,4606,4607,4608,4609,4610,4611,4612,4613,4614,4615,4616,4617,4618,4619,4620,4621,4622,4623,4624,4625,4626,4627,4628,4629,4630,4631,4632,4633,4634,4635,4636,4637,4638,4639,4640,4641,4642,4643,4644,4645,4646,4647,4648,4649,4650,4651,4652,4653,4654,4655,4656,4657,4658,4659,4660,4661,4662,4663,4664,4665,4666,4667,4668,4669,4670,4671,4672,4673,4674,4675,4676,4677,4678,4679,4680,4681,4682,4683,4684,4685,4686,4687,4688,4689,4690,4691,4692,4693,4694,4695,4696,4697,4698,4699,4700,4701,4702,4703,4704,4705,4706,4707,4708,4709,4710,4711,4712,4713,4714,4715,4716,4717,4718,4719,4720,4721,4722,4723,4724,4725,4726,4727,4728,4729,4730,4731,4732,4733,4734,4735,4736,4737,4738,4739,4740,4741,4742,4743,4744,4745,4746,4747,4748,4749,4750,4751,4752,4753,4754,4755,4756,4757,4758,4759,4760,4761,4762,4763,4764,4765,4766,4767,4768,4769,4770,4771,4772,4773,4774,4775,4776,4777,4778,4779,4780,4781,4782,4783,4784,4785,4786,4787,4788,4789,4790,4791,4792,4793,4794,4795,4796,4797,4798,4799,4800,4801,4802,4803,4804,4805,4806,4807,4808,4809,4810,4811,4812,4813,4814,4815,4816,4817,4818,4819,4820,4821,4822,4823,4824,4825,4826,4827,4828,4829,4830,4831,4832,4833,4834,4835,4836,4837,4838,4839,4840,4841,4842,4843,4844,4845,4846,4847,4848,4849,4850,4851,4852,4853,4854,4855,4856,4857,4858,4859,4860,4861,4862,4863,4864,4865,4866,4867,4868,4869,4870,4871,4872,4873,4874,4875,4876,4877,4878,4879,4880,4881,4882,4883,4884,4885,4886,4887,4888,4889,4890,4891,4892,4893,4894,4895,4896,4897,4898,4899,4900,4901,4902,4903,4904,4905,4906,4907,4908,4909,4910,4911,4912,4913,4914,4915,4916,4917,4918,4919,4920,4921,4922,4923,4924,4925,4926,4927,4928,4929,4930,4931,4932,4933,4934,4935,4936,4937,4938,4939,4940,4941,4942,4943,4944,4945,4946,4947,4948,4949,4950,4951,4952,4953,4954,4955,4956,4957,4958,4959,4960,4961,4962,4963,4964,4965,4966,4967,4968,4969,4970,4971,4972,4973,4974,4975,4976,4977,4978,4979,4980,4981,4982,4983,4984,4985,4986,4987,4988,4989,4990,4991,4992,4993,4994,4995,4996,4997,4998,4999,5000,5001,5002,5003,5004,5005,5006,5007,5008,5009,5010,5011,5012,5013,5014,5015,5016,5017,5018,5019,5020,5021,5022,5023,5024,5025,5026,5027,5028,5029,5030,5031,5032,5033,5034,5035,5036,5037,5038,5039,5040,5041,5042,5043,5044,5045,5046,5047,5048,5049,5050,5051,5052,5053,5054,5055,5056,5057,5058,5059,5060,5061,5062,5063,5064,5065,5066,5067,5068,5069,5070,5071,5072,5073,5074,

[HW:AMD:MCA:S:0:0]	n/a	AMD64 or AMD-V MCA support. Bit 0 indicates support for the MCA MSRs, bit 1 indicates support for the MCA interrupt.
[HW:AMD:SVM:S:0:0]	n/a	AMD-V support. Bit 0 indicates support for the SVM instruction set.
[HW:AMD:SVM:S:0:1:DMA_ZR_L1]	n/a	AMD-V support. Bit 1 indicates support for the DMA protection for the L1 zero page.
[HW:AMD:SVM:S:0:2:TSC_RATE_MSR]	n/a	AMD-V support. Bit 2 indicates support for the TSC rate MSR.
[HW:AMD:SVM:S:0:3:LBR_VIRT]	n/a	AMD-V support. Bit 3 indicates support for the LBR virtualization.
[HW:AMD:SVM:S:0:4:SVM_LOCK]	n/a	AMD-V support. Bit 4 indicates support for the SVM lock.
[HW:AMD:SVM:S:0:5:NRIP_SAVE]	n/a	AMD-V support. Bit 5 indicates support for the NRIP save.
[HW:AMD:SVM:S:0:6:SSSE3_MIN_DISP]	n/a	AMD-V support. Bit 6 indicates support for the SSSE3 minimal displacement.
[HW:AMD:SVM:S:0:7:AMD_VMCB_CLEAN]	n/a	AMD-V support. Bit 7 indicates support for the AMD VMCB clean.
[HW:AMD:SVM:S:0:8:FLUSH_BY_ASID]	n/a	AMD-V support. Bit 8 indicates support for the flush by ASID.
[HW:AMD:SVM:S:0:9:DECODE_ASSISTS]	n/a	AMD-V support. Bit 9 indicates support for the decode assists.
[HW:AMD:SVM:S:0:10:PAUSE_FILTER]	n/a	AMD-V support. Bit 10 indicates support for the pause filter.
[HW:AMD:SVM:S:0:11:PAUSE_FILTER_THRESH]	n/a	AMD-V support. Bit 11 indicates support for the pause filter threshold.
[HW:AMD:SVM:S:0:12:AVIC]	n/a	AMD-V support. Bit 12 indicates support for the AVIC.
[HW:AMD:SVM:S:0:13:VMSAVE_VIRT]	n/a	AMD-V support. Bit 13 indicates support for the VMSAVE virtualization.
[HW:AMD:SVM:S:0:14:VGIF]	n/a	AMD-V support. Bit 14 indicates support for the VGIF.
[HW:AMD:SVM:S:0:15:GMET]	n/a	AMD-V support. Bit 15 indicates support for the GMET.
[HW:AMD:SVM:S:0:16:SSS_CHECK]	n/a	AMD-V support. Bit 16 indicates support for the SSS check.
[HW:AMD:SVM:S:0:17:SPEC_CTRL]	n/a	AMD-V support. Bit 17 indicates support for the SPEC CTRL.
[HW:AMD:SVM:S:0:18:HOST_MCE_OVERRIDE]	n/a	AMD-V support. Bit 18 indicates support for the host MCE override.
[HW:AMD:SVM:S:0:19:TLB_CONTROL]	n/a	AMD-V support. Bit 19 indicates support for the TLB control.
[HW:AMD:SVM:S:0:20:VIRT_LOAD]	n/a	AMD-V support. Bit 20 indicates support for the virtual load.
[HW:AMD:SVM:S:0:21:VIRT_GIF]	n/a	AMD-V support. Bit 21 indicates support for the virtual GIF.
[HW:AMD:SVM:S:0:22:NPT_AVIC]	n/a	AMD-V support. Bit 22 indicates support for the NPT AVIC.
[HW:AMD:SVM:S:0:23:PMC_VIRT]	n/a	AMD-V support. Bit 23 indicates support for the PMC virtualization.
[HW:AMD:SVM:S:0:24:VNMI]	n/a	AMD-V support. Bit 24 indicates support for the VNMI.
[HW:AMD:SVM:S:0:25:VIRT_SPEC_CTRL]	n/a	AMD-V support. Bit 25 indicates support for the virtual SPEC CTRL.
[HW:AMD:SVM:S:0:26:VIRT_VMSAVE_VMLOAD]	n/a	AMD-V support. Bit 26 indicates support for the virtual VMSAVE/VMLOAD.
[HW:AMD:SVM:S:0:27:VGIF_VALUE]	n/a	AMD-V support. Bit 27 indicates support for the VGIF value.
[HW:AMD:SVM:S:0:28:SVM_PAGE_FLUSH]	n/a	AMD-V support. Bit 28 indicates support for the SVM page flush.
[HW:AMD:SVM:S:0:29:SMT_PROTECTION]	n/a	AMD-V support. Bit 29 indicates support for the SMT protection.
[HW:AMD:SVM:S:0:30:VIRT_SSBD]	n/a	AMD-V support. Bit 30 indicates support for the virtual SSBD.
[HW:AMD:SVM:S:0:31:VMCB_ADDR_CHECKS]	n/a	AMD-V support. Bit 31 indicates support for the VMCB address checks.

performance-overhead and performance-overload

Performance is a measure of how well a computer system or software does what it was intended to do. Performance-overhead and performance-overload are the amount of time and resources that are used in addition to those needed for the task itself. Common sources of overhead are overhead for memory allocation, storage or processing, or the time or cpu cycles used for a task.

performance-overhead-thread-and-performance-overload-thread

Performance-overhead-thread and performance-overload-thread are the amount of time and resources that are used in addition to those needed for the task itself, specifically for the thread. For example, thread-specific allocations, thread-specific storage or processing, or the time or cpu cycles used for a thread. Common sources of overhead are overhead for memory allocation, storage or processing, or the time or cpu cycles used for a thread. Common sources of overload are overhead for thread-specific memory allocation, thread-specific storage or processing, or the time or cpu cycles used for a thread.

The performance-overhead-thread-process-data-offset-size-time-performance-overload-thread-process-data-offset-size-time-

Appendix 2: Preference properties

This appendix provides a list of the properties in the preference properties and DHTMLpreference properties tables. Properties are listed in alphabetical order. Properties that are specific to Page Report Tools are marked with the prefix **rptpref** and properties that are specific to Page Report Profile Editor are marked with the prefix **rptprofeditor**. For more information about these properties, see the *Reference Guide*.

Properties in preference properties

Property Name	Default Value	Description
active_caching	false	Specifies whether to cache the Page Report Tools activated features.
active_engine_name	false	Specifies whether to display the Engine Name property in the Page Report Tools Properties dialog.
active_layout_name	false	Specifies whether to display the Layout Name property in the Page Report Tools Properties dialog.
active_parameters_pane	false	Specifies whether to display the Parameters pane in the Page Report Tools Properties dialog.
active_reports	true	Specifies whether to display the Reports pane in the Page Report Tools Properties dialog.
active_server_name	true	Specifies whether to display the Server Name property in the Page Report Tools Properties dialog.
active_subreport	true	Specifies whether to display the Subreport property in the Page Report Tools Properties dialog.
active_table_name	true	Specifies whether to display the Table Name property in the Page Report Tools Properties dialog.
active_tag	true	Specifies whether to display the Tag property in the Page Report Tools Properties dialog.
background_format	true	Specifies whether to display the Background Format property in the Page Report Tools Properties dialog.
cache_expiration_time	0	Specifies the cache expiration time in seconds.
completed_caching	false	Specifies whether to display the Completed Caching property in the Page Report Tools Properties dialog.
completed_completed_time	false	Specifies whether to display the Completed Completed Time property in the Page Report Tools Properties dialog.
completed_engine_status	false	Specifies whether to display the Completed Engine Status property in the Page Report Tools Properties dialog.
completed_error_message	false	Specifies whether to display the Completed Error Message property in the Page Report Tools Properties dialog.
completed_is_subreport	false	Specifies whether to display the Completed Is Subreport property in the Page Report Tools Properties dialog.
completed_layout_type	false	Specifies whether to display the Completed Layout Type property in the Page Report Tools Properties dialog.
completed_parameters	false	Specifies whether to display the Completed Parameters property in the Page Report Tools Properties dialog.
completed_report	false	Specifies whether to display the Completed Report property in the Page Report Tools Properties dialog.
completed_reports	false	Specifies whether to display the Completed Reports property in the Page Report Tools Properties dialog.
completed_subreport	false	Specifies whether to display the Completed Subreport property in the Page Report Tools Properties dialog.
completed_subreports	false	Specifies whether to display the Completed Subreports property in the Page Report Tools Properties dialog.
completed_table_name	false	Specifies whether to display the Completed Table Name property in the Page Report Tools Properties dialog.
completed_tag	false	Specifies whether to display the Completed Tag property in the Page Report Tools Properties dialog.
default_view_format	0 0 = Web 1 = Print 2 = PDF 3 = Excel 4 = Word 5 = Powerpoint 6 = Text 7 = Image 8 = Data	Specifies the default format for viewing Page Reports. Valid values are: 0 = Web 1 = Print 2 = PDF 3 = Excel 4 = Word 5 = Powerpoint 6 = Text 7 = Image 8 = Data
doc_version_number	0	Specifies the version to display the Crystal Report Document version in the Page Report Tools Properties dialog. Valid values are 0 and 1.
doc_version_output	0	Specifies the version to display the Reports Document version in the Page Report Tools Properties dialog. Valid values are 0 and 1.
enable_caching_dialog_language	true	Specifies whether to enable customizing the Caching dialog language in the Page Report Tools Properties dialog.
folder_options_dialog	true	Specifies whether to display the Folder Options dialog. This property is applied to Web parameters only.
ja_dialogs	0 0 = Automatic 1 = Japanese 2 = Non-Japanese	Specifies the Japanese dialogs to be used. Valid values are: 0 = Automatic 1 = Japanese 2 = Non-Japanese
ja_open_editing_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese dialogs to be used. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_preferences	0 0 = Automatic 1 = Japanese 2 = Non-Japanese	Specifies the Japanese dialogs to be used. Valid values are: 0 = Automatic 1 = Japanese 2 = Non-Japanese
ja_save_dialogs	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese dialogs to be used. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_save_dialogs_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese dialogs to be used. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding	0 0 = Shift_JIS 1 = EUC 2 = UTF-8	Specifies the Japanese text encoding. Valid values are: 0 = Shift_JIS 1 = EUC 2 = UTF-8
ja_text_encoding_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output_for_output	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output_for_output_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output_for_output_for_output	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output_for_output_for_output_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output_for_output_for_output_for_output	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for	0 0 = Both 1 = Page Report 2 = Dashboard	Specifies the Japanese text encoding. Valid values are: 0 = Both 1 = Page Report 2 = Dashboard
ja_text_encoding_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output_for_output

Appendix 3: System database properties in JReport Server Cluster

This appendix provides a list of properties stored in the server's system database which is created when a JReport clustered server is started, and the detailed information about these properties.

Property Name	Default Value (Range)	Description
CLUSTER		
cluster.enable_notify_server_down	false	Specifies whether to notify somebody via e-mail when a server in the cluster is down. The e-mail address is specified by the cluster.notify_server_down_address property.
cluster.notify_server_down_address	-	Specifies the e-mail addresses of the people to whom you want to send a notification e-mail when a server in the cluster is down. This property takes effect only when the cluster.enable_notify_server_down property holds a true value.
cluster.scheduler.lease.active_count	2	Specifies the active lease number in a cluster. The property value should be an integer.
cluster.scheduler.lease.check_interval	30	Specifies the time interval the cluster will use to check the number of active leases. The property value should be an integer in seconds.
cluster.scheduler.lease.enabled	false	Specifies whether to enable lease for a distributed scheduler in the cluster. If a distributed scheduler is enabled with lease, it becomes an active scheduler and will compete to trigger the schedule.
cluster.scheduler.lease.valid_time	300	Specifies the period of time that the clustered server can hold a lease before releasing it. The property value should be an integer in seconds.
cluster.share_memory.node_number	2	Specifies how many memory copies will be shared in a cluster. The property value should be a positive integer.
cluster.storage.crd_result.copy_number	2	Specifies how many copies will be made in a cluster when a new file or folder is added to the cached report data (CRD) result folder. The property value should be an integer.
cluster.storage.history.copy_number	2	Specifies how many copies will be made in a cluster when a new file or folder is added to the history folder. The property value should be an integer.
cluster.storage.realm.copy_number	2	Specifies how many copies will be made in a cluster when a new file or folder is added to the realm folder. The property value should be an integer.
LOADBALANCE		
loadbalance.type	1 (0 ~ 3)	Specifies the load-balance type. • 0 - Uses Least Current Reports Algorithm to allocate tasks. • 1 - Uses Round Robin Algorithm to allocate tasks. • 2 - Uses Least Weighted Current Reports Algorithm to allocate tasks. • 3 - Randomly allocates tasks. See Built-in load balancing algorithm for details about the four load balancing types.
loadbalance.custom_class	-	Specifies the class name of user's load balancer.
SERVER		
server.autocache.enabled	false	Specifies whether to enable to automatically cache report data. For more about cached report data, see Managing cached report data.
server.autocache.expired.time	0	Specifies the period an automatic cached data result is kept. This property takes effect when `server.autocache.enabled=true` and `server.autocache.never.expire=false`.
server.autocache.max.disk.usage	512	Specifies the maximum hard disk space for automatic cached data. The value should be between 4 MB and 1024*1024 MB. This property takes effect when server.autocache.enabled=true.
server.autocache.never.expire	true	Specifies whether automatic cached data never expires. This property takes effect when `server.autocache.enabled=true`.
server.completed.max_count	3000 (0 ~ 2147483647)	Specifies the maximum number of completed records that will be kept the Completed table. Set the value to zero if you do not want to limit the record count.
server.crd.memory.usage	4	Specifies the maximum memory usage when running a report using cached report data. The minimum value is 4 MB and can be increased up to 80% of the total Java maximum heap size (for example, -Xmx value in JRServer.bat). All cached report data being used to run reports will share the same memory space. For more about cached report data, see Managing cached report data.
server.realm.active	defaultRealm	Specifies the realm that takes effect when the server starts up. The specified realm should exist, otherwise the server will use an existing realm as active realm. The server will then record a warning message in the log file, and set the selected active realm by this property in the server.properties file.
server.security	true	Specifies whether to enable security checking on remote requests. When it is set to false, responses will be made without any security checking. This does not apply to Remote Configuration and Administration.
server.version.from.temp	false	Specifies to enable the "Publish to Version System" option when previewing a report result. This property functions the same as the Enable "Publish to Versioning System" for Background Tasks View option in the Configuration > Advanced panel of the JReport Administration page. If the value is set to true, a link Publish to Versioning System will be added to the upper-left corner of the report preview page (preview a report result using the Run command). Click the link to publish the result to version in the current format.

Properties in the following configuration files are also stored in the server's system database when a JReport clustered server is started:

File Name	Description	Mapped Server UI
mailconfig.properties	Configuration for sending e-mails	JReport Administration page > Configuration > Export > E-mail
clusterWeight.properties	Configuration for performance weight in a cluster	JReport Administration page > Cluster > Weight
LDAPProperties.xml	Configuration for LDAP support	JReport Administration page > Configuration > LDAP > Server

Setting properties by using API

For the cluster properties that are stored in the server's system database, except for loadbalance.custom_class and server.completed.max_count, all the others can be set on the JReport Administration UI. The following demo API code shows you how to set these two properties:

```
import jet.server.api.admin.ClusterAdminService;
import jet.server.api.admin.cfg.ConfigurationAdvanced;
import jet.server.api.cluster.LoadBalancer;
import jet.server.api.cluster.Member;
import jet.server.api.http.HttpRptServer;
import jet.server.api.http.HttpUtil;

public class APIDemo {

    public static void main(String[] args) {

        // Sets report home "C:\\JReport\\Server"
        System.getProperties().put("reporthome", "C:\\JReport\\Server\\");

        // Creates instance of RptServer.
        HttpUtil.initEnv(System.getProperties());
        HttpRptServer httpRptServer = HttpUtil.getHttpRptServer();
        System.out.println(" ========httpRptServer= " + httpRptServer);

        ConfigurationAdvanced configAdvanced = httpRptServer.getAdminService()
                    .getConfigService().getConfigurationAdvanced();
        //Gets the maximum number of completed records that will be kept in the Completed table.
        //Value 0 means the record number is unlimited.
        int compledRecordCount = configAdvanced.getMaxCompletedRecordCount();
        //Sets the maximum number of completed records that will be kept in the Completed table.
        configAdvanced.setMaxCompletedRecordCount(50);

        ClusterAdminService adminService = httpRptServer.getAdminService().getClusterAdminService();
        //Loads balance class
        System.out.println("load balance class" + adminService.getLoadBalanceCustomClass());
        //Sets load balance, there should be APILocalBalancer which implements interface LoadBalancer;
        adminService.setLoadBalanceCustomClass("APILocalBalancer");
    }
}
```

Appendix 4: Mapping list of UI options and properties

This appendix provides a list of Bright Server Ui options and their corresponding properties of the Bright Cluster Manager (BCM) 9.0 Ui.

The Ui options are listed in the order in which they appear in the Ui. The options are in the Bright Cluster Manager Ui, unless they are in the Bright Cluster Manager Ui Wizard, or in the Bright Cluster Manager Ui Wizard tutorial page in each case.

Ui Option	Property to set/view properties in cmsh/cmpy	Ui Option	Property to set/view properties in cmsh/cmpy
Administration Level	administrationlevel	Cluster Name	name
Firewall	firewallenabled	Time zone	time zone
Firewall Profile	firewallprofile	Time servers	timeservers
Allow Ping	allowping	Name servers	nameservers
Allow Traceroute	allowtraceroute	Search domains	searchdomains
Allow Ssh	allowssh	Name servers IP	nameserverip
Allow Ssh Port	allowsshport	Search domain	searchdomain
Allow Ssh Groups	allowsshgroups	Timeservers IP	timeserverip
Allow Ssh Networks	allowsshnetworks	Timeservers	timeservers
Allow Ssh Hosts	allowsshhosts	Timeservers Type	timeservertype
Connection Timeout	connectiontimeout	Timeservers Port	timeserverport
Kernel Module Sysctl	kernelmodulesysctl	Timeservers Allow	timeserverallow
Kernel Module	kernelmodule	Timeservers Deny	timeserverdeny
Additional Services Run	additionalservicesrun	Timeservers Options	timeserveroptions
Additional Services Stop	additionalservicesstop	Timeservers Interfaces	timeserverinterfaces
Keyboard Layout	keyboardlayout	Timeservers Allow Networks	timeserverallownetworks
Keyboard Model	keyboardmodel	Timeservers Deny Networks	timeserverdenynetworks
Keyboard Options	keyboardoptions	Timeservers Allow Hosts	timeserverallowhosts
Keyboard Toggle Options	keyboardtoggleoptions	Timeservers Deny Hosts	timeserverdenyhosts
Keyboard Terminal	keyboardterminal	Timeservers Fudge	timeserverfudge
Keyboard Unicode	keyboardunicode	Timeservers Fudge Options	timeserverfudgeoptions
Security Proxychains Proxy Servers	proxychainsproxyservers	Timeservers Keys	timeserverkeys
Security Proxychains Type	proxychainstype	Timeservers Trusted Keys	timeservertrustedkeys
Security Proxychains Port	proxychainsport	Timeservers Authkeys	timeserverauthkeys
Security Proxychains Timeout	proxychainstimeout	Timeservers Authdelay	timeserverauthdelay
Security Proxychains Random Chain	proxychainsrandomchain	Timeservers Logfile	timeserverlogfile
Security Proxychains Quiet Mode	proxychainsquietmode	Timeservers Logconfig	timeserverlogconfig
Security Proxychains Proxy Dns	proxychainsproxydns	Timeservers Statsdir	timeserverstatsdir
Security Proxychains Chain Type	proxychainschaintype	Timeservers Driftfile	timeserverdriftfile
Security Proxychains Tcp Read Time Out	proxychainstcpreadtimeout	Timeservers Leapfile	timeserverleapfile
Security Proxychains Tcp Connect Time Out	proxychainstcpconnecttimeout	Timeservers Pidfile	timeserverpidfile
Security Proxychains Chain	proxychainschain	Timeservers Ntpdate	timeserverntpdate
Security Proxychains Proxy	proxychainsproxy	Timeservers Ntpdate Options	timeserverntpdateoptions
Security Fail2ban Ignore Ip	fail2banignoreip	Timeservers Ntp Conf Extra	timeserverntpconfextra
Security Fail2ban Bantime	fail2banbantime	Timeservers Ntp Conf Extra Restrict	timeserverntpconfextrarestrict
Security Fail2ban Findtime	fail2banfindtime	Timeservers Ntp Conf Extra Server	timeserverntpconfextraserver
Security Fail2ban Maxretry	fail2banmaxretry	Timeservers Ntp Conf Extra Fudge	timeserverntpconfextrafudge
Security Fail2ban Destemail	fail2bandestemail	Timeservers Ntp Conf Extra Keys	timeserverntpconfextrakeys
Security Fail2ban Sender	fail2bansender	Timeservers Ntp Conf Extra Trusted Keys	timeserverntpconfextratrustedkeys
Security Fail2ban Action	fail2banaction	Timeservers Ntp Conf Extra Authkeys	timeserverntpconfextraauthkeys
Security Fail2ban Services	fail2banservices	Timeservers Ntp Conf Extra Logfile	timeserverntpconfextralogfile
Security Fail2ban Ignore Cmd	fail2banignorecmd	Timeservers Ntp Conf Extra Logconfig	timeserverntpconfextralogconfig
Security Fail2ban Logpath	fail2banlogpath	Timeservers Ntp Conf Extra Statsdir	timeserverntpconfextrastatsdir
Security Fail2ban Filter	fail2banfilter	Timeservers Ntp Conf Extra Driftfile	timeserverntpconfextradriftfile
Security Fail2ban Port	fail2banport	Timeservers Ntp Conf Extra Leapfile	timeserverntpconfextraleapfile
Security Fail2ban Protocol	fail2banprotocol	Timeservers Ntp Conf Extra Pidfile	timeserverntpconfextrapidfile
Security Fail2ban Maxretrydelay	fail2banmaxretrydelay	Timeservers Ntp Conf Extra Ntpdate	timeserverntpconfextrantpdate
Security Fail2ban Enabled	fail2banenabled	Timeservers Ntp Conf Extra Ntpdate Options	timeserverntpconfextrantpdateoptions
Security Fail2ban Backend	fail2banbackend	Ldap Server	ldapserver
Security Fail2ban Logtarget	fail2banlogtarget	Ldap Port	ldapport
Security Fail2ban Logmaxsize	fail2banlogmaxsize	Ldap Protocol	ldapprotocol
Security Fail2ban Logmaxfiles	fail2banlogmaxfiles	Ldap Rootdn	ldaprootdn
Security Fail2ban Logmaxdays	fail2banlogmaxdays	Ldap Rootpw	ldaprootpw
Security Fail2ban Loglevel	fail2banloglevel	Ldap Base	ldapbase
Security Fail2ban Dbfile	fail2bandbfile	Ldap Updateprovider	ldapupdateprovider
Security Fail2ban Dbpurgeage	fail2bandbpurgeage	Ldap Usetls	ldapusetls
Security Fail2ban Dbmaxmatches	fail2bandbmaxmatches	Ldap Tlsciphersuite	ldaptlsciphersuite
Security Fail2ban Dbnosynchronous	fail2bandbnosynchronous	Ldap Tlsverifycert	ldaptlsverifycert
Security Fail2ban Dbconfdir	fail2bandbconfdir	Ldap Tlscacertdir	ldaptlscacertdir
Security Fail2ban Dbconfmode	fail2bandbconfmode	Ldap Tlscacertfile	ldaptlscacertfile
Security Fail2ban Dbconfowner	fail2bandbconfowner	Ldap Tlscertdir	ldaptlscertdir
Security Fail2ban Dbconfgroup	fail2bandbconfgroup	Ldap Tlscertfile	ldaptlscertfile
Security Fail2ban Dbconfverbose	fail2bandbconfverbose	Ldap Tlskeydir	ldaptlskeydir
Security Fail2ban Jailconfprepend	fail2banjailconfprepend	Ldap Tlskeyfile	ldaptlskeyfile
Security Fail2ban Jailconfappend	fail2banjailconfappend	Ldap Tlsrandfile	ldaptlsrandfile
Security Fail2ban Jailmanagedfolder	fail2banjailmanagedfolder	Ldap Tlsecdhcurve	ldaptlsecdhcurve
Security Fail2ban Jailmanagedbuiltin	fail2banjailmanagedbuiltin	Ldap Tlsminversion	ldaptlsminversion
Security Fail2ban Jailmanagedalways	fail2banjailmanagedalways	Ldap Tlsdhmaxbits	ldaptlsdhmaxbits
Security Fail2ban Jailmanagednever	fail2banjailmanagednever	Ldap Timelimit	ldaptimelimit
Security Fail2ban Jailmanagedfolderconfoverwrite	fail2banjailmanagedfolderconfoverwrite	Ldap Bindtimelimit	ldapbindtimelimit
Security Fail2ban Jailmanagedbuiltinconfoverwrite	fail2banjailmanagedbuiltinconfoverwrite	Ldap Idletimeout	ldapidletimeout
Security Fail2ban Jailmanagedalwaysconfoverwrite	fail2banjailmanagedalwaysconfoverwrite	Ldap Writetimeout	ldapwritetimeout
Security Fail2ban Jailmanagedneverconfoverwrite	fail2banjailmanagedneverconfoverwrite	Ldap Timeout	ldaptimeout
Security Fail2ban Jailmanagedfolderfilteroverwrite	fail2banjailmanagedfolderfilteroverwrite	Ldap Bindnsupdate	ldapbindnsupdate
Security Fail2ban Jailmanagedbuiltinfilteroverwrite	fail2banjailmanagedbuiltinfilteroverwrite	Ldap Nsswitch Passwd	ldapnsswitchpasswd
Security Fail2ban Jailmanagedalwaysfilteroverwrite	fail2banjailmanagedalwaysfilteroverwrite	Ldap Nsswitch Shadow	ldapnsswitchshadow
Security Fail2ban Jailmanagedneverfilteroverwrite	fail2banjailmanagedneverfilteroverwrite	Ldap Nsswitch Group	ldapnsswitchgroup
Security Fail2ban Jailmanagedfolderactoverwrite	fail2banjailmanagedfolderactoverwrite	Ldap Nsswitch Netgroup	ldapnsswitchnetgroup
Security Fail2ban Jailmanagedbuiltinactoverwrite	fail2banjailmanagedbuiltinactoverwrite	Ldap Nsswitch Services	ldapnsswitchservices
Security Fail2ban Jailmanagedalwaysactoverwrite	fail2banjailmanagedalwaysactoverwrite	Ldap Nsswitch Ssh Pkey	ldapnsswitchsshpkey
Security Fail2ban Jailmanagedneveractoverwrite	fail2banjailmanagedneveractoverwrite	Ldap Nsswitch Sudoers	ldapnsswitchsudoers
Security Fail2ban Jailmanagedfolderfilterfailregexoverwrite	fail2banjailmanagedfolderfilterfailregexoverwrite	Ldap Nsswitch Hosts	ldapnsswitchhosts
Security Fail2ban Jailmanagedbuiltinfilterfailregexoverwrite	fail2banjailmanagedbuiltinfilterfailregexoverwrite	Ldap Nsswitch Bootparams	ldapnsswitchbootparams
Security Fail2ban Jailmanagedalwaysfilterfailregexoverwrite	fail2banjailmanagedalwaysfilterfailregexoverwrite	Ldap Nsswitch Ethers	ldapnsswitchethers
Security Fail2ban Jailmanagedneverfilterfailregexoverwrite	fail2banjailmanagedneverfilterfailregexoverwrite	Ldap Nsswitch Aliases	ldapnsswitchaliases
Security Fail2ban Jailmanagedfolderfilterignoreregexoverwrite	fail2banjailmanagedfolderfilterignoreregexoverwrite	Ldap Nsswitch Ethers Byhost	ldapnsswitchethersbyhost
Security Fail2ban Jailmanagedbuiltinfilterignoreregexoverwrite	fail2banjailmanagedbuiltinfilterignoreregexoverwrite	Ldap Nsswitch Netmasks	ldapnsswitchnetmasks
Security Fail2ban Jailmanagedalwaysfilterignoreregexoverwrite	fail2banjailmanagedalwaysfilterignoreregexoverwrite	Ldap Nsswitch Networks	ldapnsswitchnetworks
Security Fail2ban Jailmanagedneverfilterignoreregexoverwrite	fail2banjailmanagedneverfilterignoreregexoverwrite	Ldap Nsswitch Protocols	ldapnsswitchprotocols
Security Fail2ban Jailmanagedfolderfilterdatepatternoverwrite	fail2banjailmanagedfolderfilterdatepatternoverwrite	Ldap Nsswitch Rpc	ldapnsswitchrpc
Security Fail2ban Jailmanagedbuiltinfilterdatepatternoverwrite	fail2banjailmanagedbuiltinfilterdatepatternoverwrite	Ldap Nsswitch Netgroup Triple	ldapnsswitchnetgroup
Security Fail2ban Jailmanagedalwaysfilterdatepatternoverwrite	fail2banjailmanagedalwaysfilterdatepatternoverwrite	Ldap Nsswitch Initldap	ldapnsswitchinitldap
Security Fail2ban Jailmanagedneverfilterdatepatternoverwrite	fail2banjailmanagedneverfilterdatepatternoverwrite	Ldap Nsswitch Extra	ldapnsswitchextra
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Extra Order	ldapnsswitchextraorder
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Order	ldapnsswitchinitldaporder
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Criteria	ldapnsswitchinitldapcriteria
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Timeout	ldapnsswitchinitldaptimeout
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Bind Timeout	ldapnsswitchinitldapbindtimeout
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Timelimit	ldapnsswitchinitldaptimelimit
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Bind Timelimit	ldapnsswitchinitldapbindtimelimit
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Idletimeout	ldapnsswitchinitldapidletimeout
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Writetimeout	ldapnsswitchinitldapwritetimeout
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Timeout	ldapnsswitchinitldaptimeout
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Bindnsupdate	ldapnsswitchinitldapbindnsupdate
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Usetls	ldapnsswitchinitldapusetls
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Tlsciphersuite	ldapnsswitchinitldaptlsciphersuite
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Tlsverifycert	ldapnsswitchinitldaptlsverifycert
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Tlscacertdir	ldapnsswitchinitldaptlscacertdir
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Tlscacertfile	ldapnsswitchinitldaptlscacertfile
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Tlscertdir	ldapnsswitchinitldaptlscertdir
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Tlscertfile	ldapnsswitchinitldaptlscertfile
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Tlskeydir	ldapnsswitchinitldaptlskeydir
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Tlskeyfile	ldapnsswitchinitldaptlskeyfile
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Tlsrandfile	ldapnsswitchinitldaptlsrandfile
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Tlsecdhcurve	ldapnsswitchinitldaptlsecdhcurve
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Tlsminversion	ldapnsswitchinitldaptlsminversion
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Tlsdhmaxbits	ldapnsswitchinitldaptlsdhmaxbits
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Timelimit	ldapnsswitchinitldaptimelimit
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Bindtimelimit	ldapnsswitchinitldapbindtimelimit
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Idletimeout	ldapnsswitchinitldapidletimeout
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Writetimeout	ldapnsswitchinitldapwritetimeout
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Timeout	ldapnsswitchinitldaptimeout
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Bindnsupdate	ldapnsswitchinitldapbindnsupdate
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Usetls	ldapnsswitchinitldapusetls
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Tlsciphersuite	ldapnsswitchinitldaptlsciphersuite
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Tlsverifycert	ldapnsswitchinitldaptlsverifycert
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Tlscacertdir	ldapnsswitchinitldaptlscacertdir
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Tlscacertfile	ldapnsswitchinitldaptlscacertfile
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Tlscertdir	ldapnsswitchinitldaptlscertdir
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Tlscertfile	ldapnsswitchinitldaptlscertfile
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Tlskeydir	ldapnsswitchinitldaptlskeydir
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Tlskeyfile	ldapnsswitchinitldaptlskeyfile
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Tlsrandfile	ldapnsswitchinitldaptlsrandfile
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Tlsecdhcurve	ldapnsswitchinitldaptlsecdhcurve
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Tlsminversion	ldapnsswitchinitldaptlsminversion
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Tlsdhmaxbits	ldapnsswitchinitldaptlsdhmaxbits
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Timelimit	ldapnsswitchinitldaptimelimit
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Bindtimelimit	ldapnsswitchinitldapbindtimelimit
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Idletimeout	ldapnsswitchinitldapidletimeout
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Writetimeout	ldapnsswitchinitldapwritetimeout
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Timeout	ldapnsswitchinitldaptimeout
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Bindnsupdate	ldapnsswitchinitldapbindnsupdate
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Usetls	ldapnsswitchinitldapusetls
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Tlsciphersuite	ldapnsswitchinitldaptlsciphersuite
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Tlsverifycert	ldapnsswitchinitldaptlsverifycert
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Tlscacertdir	ldapnsswitchinitldaptlscacertdir
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Tlscacertfile	ldapnsswitchinitldaptlscacertfile
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Tlscertdir	ldapnsswitchinitldaptlscertdir
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Tlscertfile	ldapnsswitchinitldaptlscertfile
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Tlskeydir	ldapnsswitchinitldaptlskeydir
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Tlskeyfile	ldapnsswitchinitldaptlskeyfile
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Tlsrandfile	ldapnsswitchinitldaptlsrandfile
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Tlsecdhcurve	ldapnsswitchinitldaptlsecdhcurve
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Tlsminversion	ldapnsswitchinitldaptlsminversion
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Tlsdhmaxbits	ldapnsswitchinitldaptlsdhmaxbits
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Timelimit	ldapnsswitchinitldaptimelimit
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Bindtimelimit	ldapnsswitchinitldapbindtimelimit
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Idletimeout	ldapnsswitchinitldapidletimeout
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Writetimeout	ldapnsswitchinitldapwritetimeout
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Timeout	ldapnsswitchinitldaptimeout
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Bindnsupdate	ldapnsswitchinitldapbindnsupdate
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Usetls	ldapnsswitchinitldapusetls
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Tlsciphersuite	ldapnsswitchinitldaptlsciphersuite
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Tlsverifycert	ldapnsswitchinitldaptlsverifycert
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Tlscacertdir	ldapnsswitchinitldaptlscacertdir
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Tlscacertfile	ldapnsswitchinitldaptlscacertfile
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Tlscertdir	ldapnsswitchinitldaptlscertdir
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Tlscertfile	ldapnsswitchinitldaptlscertfile
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Tlskeydir	ldapnsswitchinitldaptlskeydir
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Tlskeyfile	ldapnsswitchinitldaptlskeyfile
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Tlsrandfile	ldapnsswitchinitldaptlsrandfile
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Tlsecdhcurve	ldapnsswitchinitldaptlsecdhcurve
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Tlsminversion	ldapnsswitchinitldaptlsminversion
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Tlsdhmaxbits	ldapnsswitchinitldaptlsdhmaxbits
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Timelimit	ldapnsswitchinitldaptimelimit
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Bindtimelimit	ldapnsswitchinitldapbindtimelimit
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Idletimeout	ldapnsswitchinitldapidletimeout
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Writetimeout	ldapnsswitchinitldapwritetimeout
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Timeout	ldapnsswitchinitldaptimeout
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Bindnsupdate	ldapnsswitchinitldapbindnsupdate
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Usetls	ldapnsswitchinitldapusetls
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Tlsciphersuite	ldapnsswitchinitldaptlsciphersuite
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Tlsverifycert	ldapnsswitchinitldaptlsverifycert
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Tlscacertdir	ldapnsswitchinitldaptlscacertdir
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Tlscacertfile	ldapnsswitchinitldaptlscacertfile
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Tlscertdir	ldapnsswitchinitldaptlscertdir
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Tlscertfile	ldapnsswitchinitldaptlscertfile
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Tlskeydir	ldapnsswitchinitldaptlskeydir
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Tlskeyfile	ldapnsswitchinitldaptlskeyfile
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Tlsrandfile	ldapnsswitchinitldaptlsrandfile
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Tlsecdhcurve	ldapnsswitchinitldaptlsecdhcurve
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Tlsminversion	ldapnsswitchinitldaptlsminversion
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Tlsdhmaxbits	ldapnsswitchinitldaptlsdhmaxbits
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Timelimit	ldapnsswitchinitldaptimelimit
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Bindtimelimit	ldapnsswitchinitldapbindtimelimit
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Idletimeout	ldapnsswitchinitldapidletimeout
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Writetimeout	ldapnsswitchinitldapwritetimeout
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Timeout	ldapnsswitchinitldaptimeout
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Bindnsupdate	ldapnsswitchinitldapbindnsupdate
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedalwaysactdelayoverwrite	Ldap Nsswitch Initldap Usetls	ldapnsswitchinitldapusetls
Security Fail2ban Jailmanagedneveractdelayoverwrite	fail2banjailmanagedneveractdelayoverwrite	Ldap Nsswitch Initldap Tlsciphersuite	ldapnsswitchinitldaptlsciphersuite
Security Fail2ban Jailmanagedfolderactdelayoverwrite	fail2banjailmanagedfolderactdelayoverwrite	Ldap Nsswitch Initldap Tlsverifycert	ldapnsswitchinitldaptlsverifycert
Security Fail2ban Jailmanagedbuiltinactdelayoverwrite	fail2banjailmanagedbuiltinactdelayoverwrite	Ldap Nsswitch Initldap Tlscacertdir	ldapnsswitchinitldaptlscacertdir
Security Fail2ban Jailmanagedalwaysactdelayoverwrite	fail2banjailmanagedal

Appendix 5: Dynamic names

When you run a report periodically, and if you publish the report result with a fixed name, then only the last generated report result can be kept on disk. In order to save the result generated every time, a dynamic result file name must be used. Similarly, you can use a dynamic directory path to avoid report management difficulties, since when a fixed directory path is specified, too many results may be generated in a single directory. JReport Server provides you with the following defined strings to form a dynamic name:

By	Strings Defined	Description (Sample)
Time		Forms the file name using the date/time when the result is generated (for example, 2006-07-25, 00:00:00 AM).
	yyyy	Returns the year (for example, 2006).
	yy	Returns the year in short form (for example, 06).
	MM	Returns the month of the year, shown in number format (for example, 12).
	dd	Returns the day of the month, (for example, 25).
	hh	Returns the hour of the day (for example, 05). It is used for the 12-hour time.
	HH	Returns the hour of the day (for example, 05). It is used for the 24-hour time.
	mm	Returns the minute of the hour (for example, 00).
	ss	Returns the second of the minute (for example, 00).
	m	Returns the AM/PM marker (for example, AM).
User		Forms the file name using user name.
	name	Returns the name of the user who is currently logged onto the computer (for example, Mike).
	z	Returns the ordinal number (for example, 001.pdf, 002.pdf, 003.pdf).
System		Forms the file name using the system variables.
	language	Returns local computer configuration language (for example, en).
	zone	Returns the system configuration time zone (for example, GMT-05.00).
	encoding	Returns the file encoding (for example, Cp1252).
	region	Returns the system configuration region (for example, US).
JReport		
	pathcat	Returns the catalog file name, with its path and extension (for example, path. SampleReports.cat).
	CAT	Returns the catalog file name, with its extension (for example, SampleReports. cat).
	cat	Returns the catalog file name, without its path and extension (for example, SampleReports).
	pathcls	Returns the report file name, with its full path (for example, catalog path and report path) and extension (for example, path1.path2.CustomerAnalysis.cls).
	CLS	Returns the report file name, with its extension (for example, CustomerAnalysis.cls).
	cls	Returns the report file name, without its extension and path (for example, CustomerAnalysis).
	sheet	Returns the report tab name in the report (for example, report1). However for formats such as .rst, .rsd, and .wst, the string "[sheet]" will be interpreted literally as "[sheet]" itself.

Naming Rules

- The defined strings are case-sensitive.
- Defined strings must be enclosed by square brackets ([ ]) which must appear in pairs, for example, [yyyy].

  If the string is not a defined string but enclosed by square brackets, an error message will be shown. For example, if you use myHTMLResult_[yy]$[YYYY][MM]M[dd]D[ABCD] $name$cls[cls].html as the result file name, you will get an error message (Invalid file name), since ABCD is not defined, but appears in square brackets.
- You should never use square brackets to enclose strings that are not defined in the above table. Strings that are not in square brackets will be directly used as (part of) the file name.
- If you want the square brackets and the strings enclosed by them to appear in the file name exactly as they are, that is, not to be parsed as defined strings, you can add a dollar sign ($) before the string. For example, if the string is $[yyyy], the result file name will be [yyyy], instead of a specific year, such as 2006.

  If you want to keep the dollar sign ($) together with the square brackets and the strings, you should add another dollar sign ($). For example, if the string is $$[m], the result file name will be $[m], and if the string is $$$$[m], the result file name will be $$$[m].
- More than one defined string can be used in the dynamic result file name, except for the ordinal number (z), such as [m][name], [yyyy].[MM].[dd], [region][yy].
- Characters are permitted to be added between different defined strings. For example, if the string is [yy]Year[MM]Month, the result could be 02Year2thMonth.
- The number of digits of the ordinal number is determined by the total number of "z"s. For example, zzz indicates 001, 002, and so on, and zz indicates 01, 02, and so on.

Notes:

- There are two differences between the naming rules of dynamic result file name and the dynamic directory name:
  - String "z" is only supported in the naming result file, but not in the naming dynamic directory. For example, `C:\folder[zz]\folder` is incorrect because the sequence for directory is not supported.
  - "/" will be replaced by "." in the naming result file. However, for naming a dynamic directory, this rule does not apply.
- When you publish a report to e-mail, string "z" is not dynamic. For example, zzz only indicates 001 and zz only indicates 01.

Here are some examples of the dynamic result file name:

The Sample Name	The Sample Result File Name
myTXTResult_[yyyy].[MM].[dd].[hh].[mm].[ss].txt	myTXTResult_2003.10.30.05.14.45.txt
myPDFResult_CT[zone][language][region]my$$[1].pdf	myPDFResult_CTGMT-05.00enUSmy$[1].pdf
myRSTResult_[pathcat][CLS].rst	myRSTResult_SampleReports.SampleReports.catCustomerAnalysis.cls.rst
myRSTResult_[zzz].rst	myRSTResult_001.rst, myRSTResult_002.rst, ...
myHTMLResult[cat]_[cls]_[zz].html	myHTMLResultSampleReports_CustomerAnalysis_01.html, myHTMLResultSampleReports_CustomerAnalysis_02.html,...
myHTMLResult_[yy]$[YYYY][MM]M[dd]D$name$cls[cls].html	myHTMLResult_02[YYYY]06M21D$name$clsCustomerAnalysis.html

Appendix 6: Parameters for running reports via URL

Running reports via URL is supported for calling JReport JSP and Servlets directly from your application or from links in JReport interactive reports. This section introduces the parameters for both ways.

JSP parameters

jrs.catalog=[catalog name with path]
jrs.report=[report name with path]
jrs.param$name=[value]
jrs.param_page=[true/false] (Default value: false. To show the parameter dialog for specifying parameter values in order to run reports, you need to set this property to true and the URL should call tryView.jsp. For details, see Specifying parameter values using dialog when running or scheduling reports via URL.)
jrs.result_type=[result type]
jrs.jdbc_driver=[JDBC driver name]
jrs.jdbc_url=[JDBC URL]
jrs.db_user=[user name]
jrs.db_pswd=[user password]
jrs.timeout_send_email=[true/false]
jrs.report_timeout=[number] (if timeout > number and jrs.timeout_send_email=true, send the email)
jrs.mailto=[email address]
jrs.mailsubject=[email subject]
jrs.mailcomments=[email content]
jrs.mailfrom=[email address of the sender]
jrs.timeout_sendmail_message=[the message which you customize]
jrs.security_file_name=[the name of a security file] (When a security file is given, the security definitions in the specified catalog will be replaced by that defined in the security file.)

- **jrs.result_type**
  Description: Specifies which format to run a report to.
  Possible values:

  0 - To Applet
  1 - To HTML
  2 - To PDF
  3 - To TEXT
  4 - To Excel
  5 - To PostScript
  6 - To Rich Text Format
  7 - To XML
  8 - To Page Report Studio (for page reports)
  8 - To Web Report Studio (for web reports)

Servlet parameters

jrs.cmd=jrs.try_vw (when the report has no parameters)
jrs.cmd=jrs.web_vw (when the report has parameters)
jrs.param$name=[value]
jrs.result_type=[result type]
jrs.jdbc_driver=[JDBC driver name]
jrs.jdbc_url=[JDBC URL]
jrs.db_user=[user name]
jrs.db_pswd=[user password]
jrs.timeout_send_email=[true/false]
jrs.report_timeout=[number] (if timeout > number and jrs.timeout_send_email=true, send the email)
jrs.mailto=[email address]
jrs.mailsubject=[email subject]
jrs.mailcomments=[email content]
jrs.mailfrom=[email address of the sender]
jrs.timeout_sendmail_message=[the message that you can customize]

- **jrs.web_vw**

  Description: Runs a report and views its result in a web browser and prompts the user if parameters are needed.
  HTTP Method: GET/POST
  URL Format: `http://HOST:PORT/jrserver/REPORT?jrs.cmd=jrs.web_vw&jrs.result_type=RESULT_TYPE&jrs.applet_type=APPLET_TYPE[_isMultiple_jrs.param$NAME=true&jrs.param$NAME=VALUE&jrs.param$NAME=VALUE...][&jrs.isjrsapltpg=TRUE/FALSE][&jrs.is_multi_files=TRUE/FALSE][&jrs.is_applet_chart=TRUE/FALSE][&jrs.has_hyperlink=TRUE/FALSE][&jrs.has_page_number=TRUE/FALSE][&jrs.use_table=TRUE/FALSE][&jrs.drilldown=TRUE/FALSE][&jrs.relative_font_size=TRUE/FALSE][&jrs.no_margin_html=TRUE/FALSE][&jrs.web_browser=BROWSER_TYPE][&jrs.no_margin_pdf=TRUE/FALSE][&jrs.is_norm_txt=TRUE/FALSE][&jrs.delimiter=DELIMITER][&jrs.udchar_width=NUMBER][&jrs.udchar_height=NUMBER][&jrs.no_margin_ps=TRUE/FALSE][&jrs.no_margin_rtf=TRUE/FALSE][&jrs.use_def_db_user=TRUE/FALSE][&jrs.db_user=DB_USER][&jrs.db_pswd=DB_PASSWORD][&jrs.jdbc_url=JDBC_URL][&jrs.jdbc_driver=JDBC_DRIVER][&jrs.rpt_language=RPT_LANGUAGE][&jrs.rpt_encoding=RPT_ENCODING][&jrs.wp=DBField='value'][&jrs.named_wp=ExistedWhereportion][&jrs.report_sheet$RPT_TAB_NAME=true]`

  URL Example: `http://localhost:8888/jrserver/SampleReports/SampleReports.cat/InvoiceReport.cls?jrs.cmd=jrs.web_vw&jrs.result_type=1&jrs.param$Country=Australia&jrs.param$State=New South Wales&jrs.param$City=Sydney&jrs.is_multi_files=true&jrs.is_applet_chart=true&jrs.use_def_db_user=false&jrs.db_user=dba&jrs.db_pswd=123456&jrs.rpt_language=en`

  Tips:

  - APPLET_TYPE:
    2 - Java Plug-In 1.2 for Windows
    3 - Java Plug-In 1.3 for Windows
  - BROWSER_TYPE:
    0 - IE or Chrome
    1 - Firefox
  - If you want to define a new WHERE portion from the URL, you can choose the property jrs.wp. For example, the report has a DBField Customer Region, and you want to restrict the field Customer Region to CA in the URL. You can then set a new WHERE portion such as, "`...jrs.result_type=1&jrs.is_applet_chart=true&jrs.wp=Customers.Region='CA'...`".
  - If you want to set an existing WHERE portion in the URL, you should use the property jrs.named_wp. For example, to set the existed WHERE portion Region from URL, use "`...jrs.result_type=1&jrs.is_applet_chart=true&jrs.named_wp=Region...`". Here Region is the WHERE portion's name.

- **jrs.try_vw**

  Description: Runs a report and views its result in a web browser when the report does not need parameters.
  HTTP Method: GET/POST
  URL Format: `http://HOST:PORT/jrserver/REPORT?jrs.cmd=jrs.try_vw&jrs.result_type=RESULT_TYPE&jrs.applet_type=APPLET_TYPE[&jrs.isjrsapltpg=TRUE/FALSE][&jrs.is_multi_files=TRUE/FALSE][&jrs.is_applet_chart=TRUE/FALSE][&jrs.has_hyperlink=TRUE/FALSE][&jrs.has_page_number=TRUE/FALSE][&jrs.use_table=TRUE/FALSE][&jrs.drilldown=TRUE/FALSE][&jrs.relative_font_size=TRUE/FALSE][&jrs.no_margin_html=TRUE/FALSE][&jrs.web_browser=BROWSER_TYPE][&jrs.no_margin_pdf=TRUE/FALSE][&jrs.is_norm_txt=TRUE/FALSE][&jrs.delimiter=DELIMITER][&jrs.udchar_width=NUMBER][&jrs.udchar_height=NUMBER][&jrs.no_margin_ps=TRUE/FALSE][&jrs.no_margin_rtf=TRUE/FALSE][&jrs.use_def_db_user=TRUE/FALSE][&jrs.db_user=DB_USER][&jrs.db_pswd=DB_PASSWORD][&jrs.jdbc_url=JDBC_URL][&jrs.jdbc_driver=JDBC_DRIVER][&jrs.rpt_language=RPT_LANGUAGE][&jrs.rpt_encoding=RPT_ENCODING][&jrs.wp=DBField='value'][&jrs.named_wp=ExistedWhereportion][&jrs.report_sheet$RPT_TAB_NAME=true]`

  URL Example: `http://localhost:8888/jrserver/SampleReports/SampleReports.cat/EmployeeInformation.cls?jrs.cmd=jrs.try_vw&jrs.result_type=1`

  Tips:

  - APPLET_TYPE:
    2 - Java Plug-In 1.2 for Windows
    3 - Java Plug-In 1.3 for Windows
  - BROWSER_TYPE:
    0 - IE or Chrome
    1 - Firefox

Appendix 7: Properties for scheduling reports via URL

This appendix presents the properties concerned with scheduling reports via URL. For the properties which are not mentioned in the following sections, you can refer to the JReport Javadoc jet.cs.util package in `<install_root>\help\api` for their details.

- General properties
- Publishing to the versioning system properties
- Publishing to the file system properties
- Publishing to e-mail properties
- Publishing to a printer properties
- Publishing to fax properties
- Publishing to FTP properties
- Time condition properties
- Notification properties
- Style group properties
- Encrypting PDF properties

General properties

When scheduling reports via URL, there are the following general properties:

- **jrs.catalog**
  Constant: jet.cs.util.APIConst.TAG_CATALOG
  Description: Specifies the catalog name with full path.
  Example: /SampleReports/SampleReports.cat, or %2fSampleReports%2fSampleReports.cat

- **jrs.report**
  Constant: jet.cs.util.APIConst.TAG_REPORT
  Description: Specifies the report name with full path.
  Example:/SampleReports/2fEmployee Information.cls, or %2fSampleReports%2fEmployee Information.cls

- **jrs.param$ + PARAMETER_NAME**
  Constant: jet.cs.util.APIConst.TAG_PARAM_PREFIX + PARAMETER_NAME
  Description: Specifies parameter values for running the report. Use jrs.param$NAME=VALUE to set parameter values of the report. Where, NAME is the parameter name and VALUE is the URL-encoded parameter value. For Example: `jrs.param$TERMSDAYS=30&jrs.param$PTODAY=May 21, 1998`. When specifying values for a multi-value parameter, you need to add _isMultiple_jrs.param$NAME=true before the parameter values to declear that the parameter supports multiple values. For example: `&_isMultiple_jrs.param$PM=true&jrs.param$PM=3&jrs.param$PM=16`.

- **jrs.param_values**
  Constant: jet.cs.util.APIConst.TAG_PARAM_VALUES
  Description: Specifies parameter values for running the report. This is the escaped string of the parameter values of the report.
  Format: PARAMETER_NAME=VALUE,PARAMETER_NAME=VALUE,...
  Examples:
  - jrs.param_values=PTODAY%3dMay%2021%5c%2c%201998%2cTERMSDAYS%3d30 (original: PTODAY=May 21\, 1998,TERMSDAYS=30)
  - jrs.param_values=STARTDATE%3d1998-05-10%2cENDDATE%3d1998-07-10 (original: STARTDATE=1998-05-10,ENDDATE=1998-07-10)

- **jrs.task_class**
  Constant: jet.cs.util.APIConst.TAG_TASK_CLASS
  Description: Specifies the class name of the task.
  Possible values:
  "jet.server.schedule.jrtasks.UpdateRptTask",
  "jet.server.schedule.jrtasks.PublishToDiskTask",
  "jet.server.schedule.jrtasks.SendJRMailTask",
  "jet.server.schedule.jrtasks.SendMailTask",
  "jet.server.schedule.jrtasks.PrintRptTask".

- **jrs.has_task_listener**
  Constant: jet.cs.util.APIConst.TAG_HAS_TASK_LISTENER
  Description: Specifies whether or not to implement TaskListener.
  Possible values: {true, false}

- **jrs.task_listener_class**
  Constant: jet.cs.util.APIConst.TAG_TASK_LISTENER_CLASS
  Description: Specifies the user's Java class name which implements the TaskListener (jrs. task_listener_class).
  Possible value: a class name (String)

- **jrs.task_id**
  Constant: jet.cs.util.APIConst.TAG_TASK_ID
  Description: Specifies the task ID of a scheduled task. This property is unnecessary if a new schedule is submitted.

- **jrs.rpt_language**
  Constant: jet.cs.util.APIConst.TAG_REPORT_LANGUAGE
  Description: Specifies the language in which to generate the report. The value you set for this command should be consistent with the language and country code part of the language property file name (for example, de_DE in ReportName_de_DE. properties). For example, if the language property file is named ReportName_de_DE. properties, you should set this command as jrs. rpt_language=de_DE. You can also use jrs.rpt_language=de&jrs.rpt_country=DE to achieve the same goal. If you only want to use jrs.rpt_language=de, then you will have to rename the language property file to ReportName_de. properties.
  Examples:
  - `http://localhost:8888/jrserver/SampleReports/SampleReports.cat/Employee Information.cls?jrs.cmd=jrs.web_vw&jrs.result_type=2&jrs.enable_nls=true&jrs.rpt_language=de_DE`
  - `http://localhost:8888/dhtmljsp/dhtml.jsp?jrs.catalog=/SampleReports/SampleReports.cat&jrs.report=/SampleReports/Employee Information.cls&jrs.enable_nls=true&jrs.rpt_language=de&jrs.rpt_country=DE`
  - If you change Employee Information_de_DE.properties to Employee Information_aaa.properties in Employee Information.cls's real path folder, for example `C:\JReport\Server\history\1\JReport_System_User457188937`, then you can use the following URL to run NLS report:

    `http://localhost:8888/dhtmljsp/dhtml.jsp?jrs.catalog=/SampleReports/SampleReports.cat&jrs.report=/SampleReports/Employee Information.cls&jrs.enable_nls=true&jrs.rpt_language=aaa`

- **jrs.rpt_encoding**
  Constant: jet.cs.util.APIConst.TAG_REPORT_ENCODING
  Description: Specifies the encoding in which to generate the report.

- **jrs.db_user**
  Constant: jet.cs.util.APIConst.TAG_DB_USER
  Description: Specifies the new database user ID if you do not want to use the default DB user ID stored in the catalog.

- **jrs.db_pswd**
  Constant: jet.cs.util.APIConst.TAG_DB_PSWD
  Description: Specifies the new database password if you do not want to use the default DB password in the catalog.

- **jrs.jdbc_url**
  Constant: jet.cs.util.APIConst.TAG_JDBC_URL
  Description: Specifies the new JDBC URL in the catalog to run a report.

- **jrs.jdbc_driver**
  Constant: jet.cs.util.APIConst.TAG_JDBC_DRIVER
  Description: Specifies the new JDBC driver in the catalog to run a report.

- **jrs.jdbc_connection_object**
  Constant: jet.cs.util.APIConst.TAG_JDBC_CONNECTION_OBJECT
  Description: Specifies to dynamically use the user's java.sql.Connection object to run a report. This parameter can be used in methods runReport(...), runReportNotWaitResult(...) and submitScheduledTask(...) of jet.server.api.RptServer.
  Example: Running a report with user's JDBC connection object by using JReport Server API.

```
//create the java.sql.Connection object.
java.sql.Connection myCon = java.sql.DriverManager.getConnection(dbUrl);
//set the parameter "jrs.jdbc_connection_object" to run a report.
propParams.put("jrs.jdbc_connection_object", myCon);
//run the report using this java.sql.Connection object instead of
//the default DB settings in the catalog.
rptServer.runReport(userID, catalog, report set, propParams);
```

- **jrs.wp**
  Constant: jet.cs.util.APIConst.TAG_WHERE_PORTION
  Description: Specifies the where portion.

- **jrs.named_wp**
  Constant: jet.cs.util.APIConst.TAG_NAMED_WHERE_PORTION
  Description: Specifies the named where portion that exists in the catalog.

- **jrs.if_print**
  Constant: jet.cs.util.APIConst.TAG_IF_PRINT
  Description: Specifies whether or not to print the report result. The default value is false

- **jrs.printer**
  Constant: jet.cs.util.APIConst.TAG_PRINTER
  Description: Specifies the printer for printing the report result.

- **jrs.mail_to_referuser**
  Constant: jet.cs.util.APIConst.TAG_MAIL_TO_REFERUSER
  Description: Specifies whether or not to send mails to each user who is authorized to view a report with cached report bursting.
  Possible values: {true, false}

Publishing to the versioning system properties

When scheduling reports via URL, the publishing to the versioning system properties are as follows:

- **jrs.expire_days**
  Constant: jet.cs.util.APIConst.TAG_EXPIRE_DAYS
  Description: Specifies the number of days a result version will be kept until it expires. The default value is 30.

Publishing to the file system properties

When scheduling reports via URL, the publishing to the file system properties are as follows:

- **jrs.rst_dir**
  Constant: jet.cs.util.APIConst.TAG_RST_DIR
  Description: Specifies the path for the published JReport result file.
  Possible values: /publish_site
- **jrs.rst**
  Constant: jet.cs.util.APIConst.TAG_RST
  Description: Specifies the result filename for the published JReport result file.
  Possible values: Employee Information.rst
- **jrs.html_dir**
  Constant: jet.cs.util.APIConst.TAG_HTML_DIR
  Description: Specifies the path for the published HTML file.
  Possible values: /publish_site
- **jrs.html**
  Constant: jet.cs.util.APIConst.TAG_HTML
  Description: Specifies the file name for the published HTML file.
  Possible values: Employee Information.htm
- **jrs.is_dhtml**
  Constant: APIConst.TAG_IS_DHTML
  Description: Specifies to generate a TOC (To HTML).
  Possible values: {true, false}. The default is false.
- **jrs.is_multi_files**
  Constant: jet.cs.util.APIConst.TAG_IS_MULTI_FILES
  Description: Specifies whether HTML is generated to a single or to multiple files (To HTML).
  Possible values: {true, false}. The default is true.
- **jrs.is_applet_chart**
  Constant: jet.cs.util.APIConst.TAG_IS_CHART_APPLET
  Description: Specifies that the chart in HTML file is an applet (To HTML).
  Possible values: {true, false}. The default is true.
- **jrs.has_hyperlink**
  Constant: jet.cs.util.APIConst.TAG_HAS_HYPERLINK
  Description: Specifies to generate hyperlinks in the HTML file (To HTML).
  Possible values: {true, false}. The default is true.
- **jrs.has_page_number**
  Constant: jet.cs.util.APIConst.TAG_HAS_PAGE_NUMBER
  Description: Specifies to generate page numbers (To HTML).
  Possible values: {true, false}. The default is true.
- **jrs.use_table**
  Constant: jet.cs.util.APIConst.TAG_USE_TABLE
  Description: Specifies to use the HTML table to generate the HTML page (To HTML).
  Possible values: {true, false}. The default is false
- **jrs.relative_font_size**
  Constant: jet.cs.util.APIConst.TAG_RELATIVE_FONT_SIZE
  Description: Specifies to generate HTML pages with a relative font size (To HTML).
  Possible values: {true, false}. The default is false.
- **jrs.drilldown**
  Constant: jet.cs.util.APIConst.TAG_DRILLDOWN
  Description: Specifies to drilldown on the HTML page (To HTML).
  Possible values: {true, false}. The default is false.
- **jrs.no_margin_html**
  Constant: jet.cs.util.APIConst.TAG_NO_MARGIN_HTML
  Description: Specifies that there is no margin in the HTML page (To HTML).
  Possible values: {true, false}. The default is false.
- **jrs.web_browser**
  Constant: jet.cs.util.APIConst.TAG_WEB_BROWSER
  Description: Specifies which web browser to generate the HTML pages for (To HTML).
  Possible values: {0, 1}. 0 - IE or Chrome, 1 - Firefox. The default is 0.
- **jrs.pdf_dir**
  Constant: jet.cs.util.APIConst.TAG_PDF_DIR
  Description: Specifies the path of the published PDF file.
  Possible values: /publish_site
- **jrs.pdf**
  Constant: jet.cs.util.APIConst.TAG_PDF
  Description: Specifies the file name of the published PDF file.
  Possible values: Employee Information.rst
- **jrs.no_margin_pdf**
  Constant: jet.cs.util.APIConst.TAG_NO_MARGIN_PDF
  Description: Specifies to display no margin (To PDF).
  Possible values: {true, false}. The default is false.
- **jrs.compress_image_pdf**
  Constant: jet.cs.util.APIConst.TAG_COMPRESS_IMAGE_PDF
  Description: Specifies the compression ratio of the image in the report (To PDF).
  Possible values: 1 to 100. The default is 20.
- **jrs.print_mode_pdf**
  Constant: jet.cs.util.APIConst.TAG_PRINT_MODE_PDF
  Description: Specifies whether to use Generate charts and barcodes using images (recommended) (false) or Generate charts and barcodes using vector graphics (true) (To PDF). By checking Generate charts and barcodes using vector graphics, the exporting speed is much improved.
  Possible values: {true, false}. The default is false.
- **jrs.text_dir**
  Constant: jet.cs.util.APIConst.TAG_TEXT_DIR
  Description: Specifies the path of the published TXT file.
  Possible values: /publish_site
- **jrs.text**
  Constant: jet.cs.util.APIConst.TAG_TEXT
  Description: Specifies the file name of the published TXT file.
  Possible values: Employee Information.txt
- **jrs.is_norm_txt**
  Constant: jet.cs.util.APIConst.TAG_IS_NORMAL_TEXT
  Description: Specifies whether this text file is in Delimited Format or not (To TEXT). When it is set to false, the output format is Delimited Format.
  Possible values: {true, false} The default is true.
- **jrs.is_text_delimiter**
  Constant: jet.cs.util.APIConst.TAG_TEXT_IS_DELIMITER
  Description: Specify a delimiter to separate fields (To TEXT).
  Possible values: It has three values:
  - **jrs.delimiter**
    Constant: jet.cs.util.APIConst.TAG_DELIMITER
    Description: If jrs.is_text_delimiter=jrs.delimiter, you can use jrs.delimiter to specify a delimiter character, for example, jrs.delimiter=','.
    Possible values: A user defined delimiter.
  - **jrs.is_tab**
    Constant: jet.cs.util.APIConst.TAG_IS_TAB
    Description: Uses a Tab delimiter to separate the fields.
  - **jrs.is_csv**
    Constant: jet.cs.util.APIConst.TAG_IS_CSV
    Description: Exports to the CSV format text file.
- **jrs.is_repeat**
  Constant: jet.cs.util.APIConst.TAG_IS_REPEAT
  Description: Specifies whether the same content is repeated (To TEXT).
  Possible values: {true, false}. The default is false.
- **jrs.udchar_width**
  Constant: jet.cs.util.APIConst.TAG_CHAR_WIDTH
  Description: Specifies the user defined width. It would be given up if the value is 0 or less than 0 (To TEXT).
- **jrs.udchar_height**
  Constant: jet.cs.util.APIConst.TAG_CHAR_HEIGHT
  Description: Specifies the user defined height. It would be given up if the value is 0 or less than 0 (To TEXT).
- **jrs.is_quotemark**
  Constant: jet.cs.util.APIConst.TAG_IS_QUOTEMARK
  Description: Specifies that quotation marks should be placed in the published text file.
  Possible values: {true, false}. The default is true.
- **jrs.hasHeadFoot**
  Constant: jet.cs.util.APIConst.TAG_HASHEADFOOT
  Description: Specifies to generate the page header or footer (To TEXT).
  Possible values: {true, false}. The default is true.
- **jrs.txt_compress**
  Constant: jet.cs.util.APIConst.TAG_TXT_COMPRESS
  Description: Specifies to compress the published text file.
  Possible values: {true, false}. The default is false.
- **jrs.excel_dir**
  Constant: jet.cs.util.APIConst.TAG_EXCEL_DIR
  Description: Specifies the path of the published Excel file.
  Possible values: /publish_site
- **jrs.excel**
  Constant: jet.cs.util.APIConst.TAG_EXCEL
  Description: Specifies the file name of the published Excel file.
  Possible values: Employee Information.rst
- **jrs.ps_dir**
  Constant: jet.cs.util.APIConst.TAG_PS_DIR
  Description: Specifies the path of the published PostScript file.
  Possible values: /publish_site
- **jrs.ps**
  Constant: jet.cs.util.APIConst.TAG_PS
  Description: Specifies the result file name of the published PostScript file.
  Possible values: Employee Information.rst
- **jrs.no_margin_ps**
  Constant: jet.cs.util.APIConst.TAG_NO_MARGIN_PS
  Description: Specifies that no margins are included (To PostScript).
  Possible values: {true, false}. The default is false.
- **jrs.rtf_dir**
  Constant: jet.cs.util.APIConst.TAG_PS_DIR
  Description: Specifies the path for publishing to Rich Text Format.
  Possible values: /publish_site
- **jrs.rtf**
  Constant: jet.cs.util.APIConst.TAG_RTF
  Description: Specifies the result file name for publishing to Rich Text Format.
  Possible values: Employee Information.rst
- **jrs.no_margin_rtf**
  Constant: jet.cs.util.APIConst.TAG_NO_MARGIN_RTF
  Description: Specifies that no margins are included (To Rich Text Format).
  Possible values: {true, false}. The default is false.
- **jrs.xml_dir**
  Constant: jet.cs.util.APIConst.TAG_XML_DIR
  Description: Specifies the path for publishing to XML file.
  Possible values: /publish_site
- **jrs.xml**
  Constant: jet.cs.util.APIConst.TAG_XML
  Description: Specifies the result file name for publishing to XML file.
  Possible values: Employee Information.xml
- **jrs.is_only_data**
  Constant: jet.cs.util.APIConst.TAG_IS_ONLY_DATA
  Description: Specifies to contain only data in the published XML file.
  Possible values: {true, false}. The default is false.
- **jrs.to_disk**
  Constant: jet.cs.util.APIConst.TAG_TO_DISK
  Description: Specifies that the report is scheduled for publishing to disk.
  Possible values: {true, false}
- **jrs.to_disk_???_path_type**
  ??? specifies the format of the published report. For example, HTML or PDF. See the example below.
- **jrs.to_disk_pdf_path_type**
  Constant: jet.cs.util.APIConst.TAG_TO_DISK_PDF_PATH_TYPE
  Description: Specifies whether to schedule the report for publishing to a real disk path or to the JReport Server resource path in PDF format.
  Possible values: {0, 1}. 0 - publish to the JReport Server resource path, 1 - publishing to a real disk path

Examples

Example 1: Publish a report to a real disk path in PDF format

```
http://localhost:8888/jinfonet/submitSchedPage.jsp?jrs.cmd=jrs.submit_schedule&jrs.
catalog=%2fSampleReports%2fSampleReports.cat&jrs.report=%2fSampleReports%2fEmployee
Information.cls&jrs.task_class=jet.server.schedule.jrtasks.PublishRptTask&jrs.
to_disk=true&jrs.to_disk_pdf_path_type=1&jrs.result_type=2&jrs.auth_uid=admin&jrs.
auth_pwd=admin&jrs.pdf_dir=d:\&jrs.pdf=abc.pdf&jrs.to_pdf=true
```

Example 2: Publish a report to a real disk path in TXT format

```
http://localhost:8888/jrserver?jrs.cmd=jrs.submit_schedule&jrs.catalog=%2fSampleReports%
2fSampleReports.cat&jrs.report=%2fSampleReports%2fEmployee Information.cls&jrs.
task_class=jet.server.schedule.jrtasks.PublishRptTask&jrs.to_disk=true&jrs.
to_disk_text_path_type=1&jrs.result_type=3&jrs.auth_uid=admin&jrs.auth_pwd=admin&jrs.
text_dir=d:\&jrs.text=abc.txt&jrs.to_text=true
```

Publishing to e-mail properties

When scheduling reports via URL, the publishing to e-mail properties are as follows:

- **jrs.jrmail + NUMBER**
  Constant: jet.cs.util.APIConst.TAG_JRMAIL + NUMBER
  Description: Contains specifications (JReport mail properties) of one send-mail task for a report.
  Example: `jrs.jrmail0=jrs.mailsubject%3dreport%2520Employee Information%26jrs.mailformat%3d0%26jrs.mailto%3dreader1%26jrs.jrmail1=jrs.mailsubject%3dreport%2520Employee Information%26jrs.mailformat%3d4%26jrs.mailto%3dreader2%252c%2520reader3%26`

- **jrs.csmail + NUMBER**
  Constant: jet.cs.util.APIConst.TAG_NORMAL_MAIL + NUMBER
  Description: Contains specifications (JReport mail properties) of one send-mail task for sending an e-mail with or without an attachment. This property is used via URL or via invoking Server API.
  Example: `jrs.csmail0=jrs.mailcomments%3dto%2520user1.%26jrs.mailsubject%3dtest%2520sending%2520normal%2520emails%2520without%2520report%26jrs.mailto%3dtim%40support.jinfonet.com&jrs.mailattach1%3dd%3a%5ctest1.txt&jrs.mailattach0%3dd%3a%5ctest0.txtjrs.csmail1=jrs.mailcomments%3dto%2520user2.%26jrs.mailsubject%3dtest%2520sending%2520normal%2520emails%2520without%2520report%26jrs.mailto%3dhelen%40support.jinfonet.com%26jrs.mailattach1%3dd%3a%5ctest2.txt%26jrs.mailattach0%3dd%3a%5ctest3.txt`

Properties in jrs.jrmail

- **jrs.mailto**
  Constant: jet.cs.util.APIConst.TAG_MAIL_TO
  Description: Specifies the mail address.

- **jrs.mailcc**
  Constant: jet.cs.util.APIConst.TAG_MAIL_CC
  Description: Specifies the mail address to be carbon copied to.

- **jrs.mailbcc**
  Constant: jet.cs.util.APIConst.TAG_MAIL_BCC
  Description: Specifies the mail address to be blind carbon copied to.

- **jrs.mailsubject**
  Constant: jet.cs.util.APIConst.TAG_MAIL_SUBJECT
  Description: Specifies the "SUBJECT" of the e-mail.

- **jrs.mailcomments**
  Constant: jet.cs.util.APIConst.TAG_MAIL_COMMENT
  Description: Specifies the "COMMENTS" field of the e-mail.

- **jrs.mailformat**
  Constant: jet.cs.util.APIConst.TAG_MAIL_FORMAT
  Description: Specifies the mail format.
  Possible values: {0, 1, 2, 3, 4, 6, 7, 8, 9, 11}. The default value is 0.
  - 0 - E-mail Result in HTML E-mail Format
  - 1 - E-mail Result in Plain Text E-mail Format
  - 2 - Attachment in HTML Format
  - 3 - Attachment in PDF Format
  - 4 - Attachment in JReport Result Format
  - 6 - Attachment in PostScript Format
  - 7 - Attachment in Excel Format
  - 8 - Attachment in RTF Format
  - 9 - Attachment in XML Format
  - 11 - Attachment in Text Format

- **jrs.mailcompress**
  Constant: jet.cs.util.APIConst.TAG_MAIL_COMPRESS
  Description: Specifies whether to enable Java archive compress.
  Possible values: {true, false}. The default is false.

- **jrs.udchar_width**
  Constant: jet.cs.util.APIConst.TAG_CHAR_WIDTH
  Description: Specifies the user defined character width. It has no effect if the value is 0 or less than 0 (for TEXT format).

- **jrs.udchar_height**
  Constant: jet.cs.util.APIConst.TAG_CHAR_HEIGHT
  Description: Specifies the user defined character height. It has no effect if the value is 0 or less than 0 (for TEXT format).

- **jrs.no_margin_mail**
  Constant: jet.cs.util.APIConst.TAG_NO_MARGIN_MAIL
  Description: Specifies that there are no margins for the report in the e-mail.
  Possible values: {true, false}. The default is false.

- **jrs.use_table_mail**
  Constant: jet.cs.util.APIConst.TAG_USE_TABLE_MAIL
  Description: Specifies to use the HTML table to generate an HTML format e-mail.
  Possible values: {true, false}. The default is false, which means to generate the HTML table.

- **jrs.mailattach + NUMBER**
  Constant: jet.cs.util.APIConst.TAG_MAIL_ATTACH
  Description: Specifies the attached file for this mail. You can attach multiple files with one mail.
  Possible values: The attached file name
  Example: `jrs.mailattach0=d%3a%5ctest1.txt (where "d%3a%5ctest1.txt" is the URL-encoded string of "d:\test1.txt")`

- **jrs.mailencoding**
  Constant: jet.cs.util.APIConst.TAG_MAIL_ENCODING
  Description: Specifies the encoding of the mails.
  Possible values: UTF-8, UTF-16, ISO8859-1 and so on.

  Note: jrs.mailencoding is used to specify the mail encoding in the URL. When sending mails by RMI API, sometimes, wrong characters may be returned in the mail. In order to avoid such problems, specify the same correct value of -Djreport.url.encoding on both the server and RMI client side.

  For example, your web application calls JReport Server (standalone) via the RMI function from WebSphere, and UTF-8 (use UTF-8 rather than UTF8) will be used as the mail encoding, do as follows:

  - Specify -Djreport.url.encoding=UTF-8 for both JVM running JReport Server and WebSphere.
  - Specify jrs.mailencoding=UTF-8.

Example

```
http://localhost:8888/jinfonet/submitSchedPage.jsp?jrs.cmd=jrs.submit_schedule&jrs.catalog=/SampleReports/SampleReports.cat&jrs.report=/SampleReports/CustomerAnalysis.cls&jrs.param$P_StartDate=01/01/2006&jrs.param$p_EndDate=12/31/2007&jrs.task_class=jet.server.schedule.jrtasks.SendMailTask&jrs.to_mail=true&jrs.csmail0=jrs.mailto%3dxxx@jinfonet.com%26jrs.mailcomments%3dtouser1.%26jrs.mailsubject%3dreportkkkk%26jrs.mailattach0%3dd%3a%5ctemp%5ctest1.txt%26jrs.mailattach1%3dd%3a%5ctemp%5ctest2.txt
```

Publishing to a printer properties

When scheduling reports via URL, the publishing to a printer properties are as follows:

- **jrs.printer**
  Constant: jet.cs.util.APIConst.TAG_PRINTER
  Description: Specifies the printer with which to print the report result.

- **jrs.print_copies**
  Constant: jet.cs.util.PrintConst.TAG_PRINT_COPIES
  Description: Specifies the number of copies to print the report result.

- **jrs.has_margins**
  Constant: jet.cs.util.PrintConst.TAG_HAS_MARGINS
  Description: Specifies whether to have margins in the printed report result.

  When jrs.has_margins is true, the following properties are available for use:

  - **jrs.margins_left**
    Constant: jet.cs.util.PrintConst.TAG_MARGINS_LEFT
    Description: Specifies the length of the left margin to print the report result.

  - **jrs.margins_right**
    Constant: jet.cs.util.PrintConst.TAG_MARGINS_RIGHT
    Description: Specifies the length of the right margin to print the report result.

  - **jrs.margins_top**
    Constant: jet.cs.util.PrintConst.TAG_MARGINS_TOP
    Description: Specifies the length of the top margin to print the report result.

  - **jrs.margins_bottom**
    Constant: jet.cs.util.PrintConst.TAG_MARGINS_BOTTOM
    Description: Specifies the length of the bottom margin to print the report result.

  - **jrs.margins_unit**
    Constant: jet.cs.util.PrintConst.TAG_MARGINS_UNIT
    Description: Specifies the unit to print the report result.
    Values: There are two values:

    - **jrs.margins_unit_mm**
      Constant: jet.cs.util.PrintConst.MARGINS_UNIT_MM
      Description: Specifies mm as the margin unit.

    - **jrs.margins_unit_inch**
      Constant: jet.cs.util.PrintConst.MARGINS_UNIT_INCH
      Description: Specifies inch as the margin unit.

- **jrs.print_mediatray**
  Constant: jet.cs.util.PrintConst.TAG_PRINT_MEDIATRAY
  Description: Specifies the custom tray to put the printing paper.
  Example:
  `http://localhost:8888/jrserver?jrs.cmd=jrs.submit_schedule&jrs.catalog=%2fSampleReports%2fSampleReports.cat&jrs.report=%2fSampleReports%2fBanded_Link.cls&jrs.task_class=jet.server.schedule.jrtasks.PublishRptTask&jrs.launch_type=0&jrs.to_printer=true&jrs.print_usejdk=2&jrs.printer=smartprint&jrs.print_copies=2&jrs.print_mediatray=Form-Source&jrs.has_margins=true&jrs.margins_left=8&jrs.margins_right=9&jrs.margins_top=7&jrs.margins_bottom=7&jrs.margins_unit=jrs.margins_unit_mm`

  > **Note:** In this case, you must set the value for the following properties at the same time: jrs.margins_left, jrs.margins_right, jrs.margins_top, jrs.margins_bottom and jrs.margins_unit; otherwise, it can not work smoothly.

Publishing to fax properties

When scheduling reports via URL, the publishing to fax properties are as follows:

- **jrs.to_fax**
  Constant: jet.cs.util.APIConst.TAG_TO_FAX
  Description: Specifies whether to publish the report result to fax.
  Possible values: {true, false}
  - **jrs.to_fax_quality**
    Constant: jet.cs.util.APIConst.TAG_TO_FAX_QUALITY
    Description: Specifies the fax quality.
    Possible values: {jrs.to_fax_quality_best, jrs.to_fax_quality_fast, jrs.to_fax_quality_normal}
    - **jrs.to_fax_quality_best**
      Constant: jet.cs.util.APIConst.TAG_TO_FAX_QUALITY_BEST
      Description: Indicates the best fax quality.
    - **jrs.to_fax_quality_fast**
      Constant: jet.cs.util.APIConst.TAG_TO_FAX_QUALITY_FAST
      Description: Indicates the fast fax quality.
    - **jrs.to_fax_quality_normal**
      Constant: jet.cs.util.APIConst.TAG_TO_FAX_QUALITY_NORMAL
      Description: Indicates the normal fax quality.
  - **jrs.to_fax_is_inclue_cover**
    Constant: jet.cs.util.APIConst.TAG_TO_FAX_IS_INCLUDE_COVER
    Description: Specifies whether to send a cover sheet with the fax.
    Possible values: {true, false}
    - **jrs.to_fax_date**
      Constant: jet.cs.util.APIConst.TAG_TO_FAX_DATE
      Description: Specifies the date on which the fax is to be sent.
    - **jrs.to_fax_to**
      Constant: jet.cs.util.APIConst.TAG_TO_FAX_TO
      Description: Specifies the fax recipient.
    - **jrs.to_fax_to_fax_num**
      Constant: jet.cs.util.APIConst.TAG_TO_FAX_TO_FAX_NUMBER
      Description: Specifies the fax number of the recipient.
  - **jrs.to_fax_from**
    Constant: jet.cs.util.APIConst.TAG_TO_FAX_FROM
    Description: Specifies the fax sender.
  - **jrs.to_fax_from_company_name**
    Constant: jet.cs.util.APIConst.TAG_TO_FAX_FROM_COMPANY_NAME
    Description: Specifies the sender's company.
  - **jrs.to_fax_from_phone**
    Constant: jet.cs.util.APIConst.TAG_TO_FAX_FROM_PHONE
    Description: Specifies the sender's phone number.
  - **jrs.to_fax_subject**
    Constant: jet.cs.util.APIConst.TAG_TO_FAX_SUBJECT
    Description: Specifies the subject of the fax.
  - **jrs.to_fax_comments**
    Constant: jet.cs.util.APIConst.TAG_TO_FAX_COMMENTS
    Description: Specifies the comments of the fax.
  - **jrs.to_fax_urgent**
    Constant: jet.cs.util.APIConst.TAG_TO_FAX_URGENT
    Description: Specifies whether the fax is urgent.
    Possible values: {true, false}
  - **jrs.to_fax_for_review**
    Constant: jet.cs.util.APIConst.TAG_TO_FAX_FOR_REVIEW
    Description: Specifies whether the fax is for review.
    Possible values: {true, false}
  - **jrs.to_fax_please_comment**
    Constant: jet.cs.util.APIConst.TAG_TO_FAX_PLEASE_COMMENT
    Description: Specifies whether the recipient is required to comment on the content of the fax.
    Possible values: {true, false}
  - **jrs.to_fax_please_reply**
    Constant: jet.cs.util.APIConst.TAG_TO_FAX_PLEASE_REPLY
    Description: Specifies whether a reply is required for the fax.
    Possible values: {true, false}

Publishing to FTP properties

When scheduling reports via URL, the publishing to FTP properties are as follows:

- **jrs.to_FTP**
  Constant: jet.cs.util.APIConst.TAG_TO_FTP
  Description: Specifies whether to publish the report result to FTP.
  Possible values: {true, false}
- **jrs.ftp**
  Constant: jet.cs.util.APIConst.TAG_JRFTP
  Description: The JReport FTP. It is used to parse multiple sites.
- **jrs.ftpLbl**
  Constant: jet.cs.util.APIConst.TAG_FTP_LBL
  Description: Specifies the label of the FTP server.
  Possible values: Jinfonet FTP Site
- **jrs.ftpHost**
  Constant: jet.cs.util.APIConst.TAG_FTP_HOST
  Description: Specifies the host of the FTP server.
  Possible values: 192.168.126.1
- **jrs.ftpPort**
  Constant: jet.cs.util.APIConst.TAG_FTP_PORT
  Description: Specifies the port of the FTP server.
  Possible values: 21
- **jrs.ftpUn**
  Constant: jet.cs.util.APIConst.TAG_FTP_UN
  Description: Specifies the user name for logging in the FTP server.
  Possible values: FTP user1
- **jrs.ftpPsd**
  Constant: jet.cs.util.APIConst.TAG_FTP_PSD
  Description: Specifies the password for logging in the FTP server.
  Possible values: 123456
- **jrs.ftpAcct**
  Constant: jet.cs.util.APIConst.TAG_FTP_ACCT
  Description: Specifies the account for logging in the FTP server.
- **jrs.ftpLoc**
  Constant: jet.cs.util.APIConst.TAG_FTP_LOC
  Description: Specifies the remote directory on the FTP server to which the files will be published.
  Possible values: Jreport/FTP/20061218/
- **jrs.ftpHdlCls**
  Constant: jet.cs.util.APIConst.TAG_FTP_HANDLER_CLASS
  Description: Specifies the FTP client-end handler class for communicating with the FTP server.
  Possible values: FTPHandler class name or null which is the default value.
- **jrs.ftpProt**
  Constant: jet.cs.util.APIConst.TAG_FTP_PROT
  Description: Specifies the protocol for communicating with the FTP server.
  Possible values: {0, 1, 2, 3}. 0 - FTP, 1 - SFTP, 2 - SCP, 3 - FTPS
- **jrs.ftpsConType**
  Constant: jet.cs.util.APIConst.TAG_FTPS_CONN_TYPE
  Description: Specifies the connection type of FTPS.
  Possible values: {0, 1}. 0 - EXPLICIT, 1 - IMPLICIT
- **jrs.ftpsEnNoSec**
  Constant: jet.cs.util.APIConst.TAG_FTPS_EN_NO_SEC
  Description: Specifies whether to enable falling back to the no-security FTP connection if the explicit FTPS connection is not available.
  Possible values: {true, false}
- **jrs.ftpsKSType**
  Constant: jet.cs.util.APIConst.TAG_FTPS_KS_TYPE
  Description: Specifies the keystore type of FTPS.
- **jrs.ftpsKSFile**
  Constant: jet.cs.util.APIConst.TAG_FTPS_KS_FILE
  Description: Specifies the keystore file of FTPS.
- **jrs.ftpsKSPsd**
  Constant: jet.cs.util.APIConst.TAG_FTPS_KS_PSD
  Description: Specifies the keystore password of FTPS.
- **jrs.ftpsKMAlg**
  Constant: jet.cs.util.APIConst.TAG_FTPS_KM_ALG
  Description: Specifies the keymanager algorithm of FTPS.
- **jrs.ftpsSecProt**
  Constant: jet.cs.util.APIConst.TAG_FTPS_SEC_PROT
  Description: Specifies the security protocol of FTPS.
- **jrs.ftpsTMAlg**
  Constant: jet.cs.util.APIConst.TAG_FTPS_TM_ALG
  Description: Specifies the trustmanager algorithm of FTPS.
- **jrs.ftpsTransMode**
  Constant: jet.cs.util.APIConst.TAG_FTPS_TRANS_MODE
  Description: Specifies the transfer mode of FTPS.
- **jrs.ftpsTSType**
  Constant: jet.cs.util.APIConst.TAG_FTPS_TS_TYPE
  Description: Specifies the truststore type of FTPS.
- **jrs.ftpsTSFile**
  Constant: jet.cs.util.APIConst.TAG_FTPS_TS_FILE
  Description: Specifies the truststore file of FTPS.
  Possible values: truststore_file_name
- **jrs.ftpsTSPsd**
  Constant: jet.cs.util.APIConst.TAG_FTPS_TS_PSD
  Description: Specifies the truststore password of FTPS.
- **jrs.sftpC2SCmpA**
  Constant: jet.cs.util.APIConst.TAG_SFTP_C2S_CmpA
  Description: Specifies the C2S compression algorithms of SFTP/SCP.
- **jrs.sftpC2SCphA**
  Constant: jet.cs.util.APIConst.TAG_SFTP_C2S_CphA
  Description: Specifies the C2S cipher algorithms of SFTP/SCP.
- **jrs.sftpC2SLang**
  Constant: jet.cs.util.APIConst.TAG_SFTP_C2S_Lang
  Description: Specifies the C2S language of SFTP/SCP.
- **jrs.sftpC2SMA**
  Constant: jet.cs.util.APIConst.TAG_SFTP_C2S_MA
  Description: Specifies the C2S MAC algorithms of SFTP/SCP.
- **jrs.sftpHKAlgs**
  Constant: jet.cs.util.APIConst.TAG_SFTP_HKAlgs
  Description: Specifies the host key algorithms of SFTP/SCP.
- **jrs.sftpKexAlgs**
  Constant: jet.cs.util.APIConst.TAG_SFTP_KexAlgs
  Description: Specifies the kex algorithms of SFTP/SCP.
- **jrs.sftpKH**
  Constant: jet.cs.util.APIConst.TAG_SFTP_KNOWNHOSTS
  Description: Specifies the knownhosts file of SFTP/SCP.
- **jrs.sftpS2CCmpA**
  Constant: jet.cs.util.APIConst.TAG_SFTP_S2C_CmpA
  Description: Specifies the s2c compression algorithms of SFTP/SCP.
- **jrs.sftpS2CCphA**
  Constant: jet.cs.util.APIConst.TAG_SFTP_S2C_CphA
  Description: Specifies the S2C cipher algorithms of SFTP/SCP.
- **jrs.sftpS2CLang**
  Constant: jet.cs.util.APIConst.TAG_SFTP_S2C_Lang
  Description: Specifies the S2C language of SFTP/SCP.
- **jrs.sftpS2CMA**
  Constant: jet.cs.util.APIConst.TAG_SFTP_S2C_MA
  Description: Specifies the S2C MAC algorithms of SFTP/SCP.
- **jrs.sftpSHKC**
  Constant: jet.cs.util.APIConst.TAG_SFTP_SHKC
  Description: Specifies whether to check the strict host key.
  Possible values: {yes, no}
- **jrs.ftp_param_validation**
  Constant: jet.cs.util.APIConst.CMD_FTP_PARAM_VALIDATION
  Description: Specifies the command of checking the validation of FTP connection options.
  Possible values: The following four properties
  - **TAG_FTP_CONNECTION_FAILED**
    Constant: jet.cs.util.APIConst.TAG_FTP_CONNECTION_FAILED
    Description: The connection cannot be created because the host name/IP or port is not valid.
    Possible values: {100} (the only value)
  - **TAG_FTP_CONNECTION_IS_OK**
    Constant: jet.cs.util.APIConst.TAG_FTP_CONNECTION_IS_OK
    Description: The connection is valid.
    Possible values: {200} (the only value)
  - **TAG_FTP_NO_PERMISSION**
    Constant: jet.cs.util.APIConst.TAG_FTP_NO_PERMISSION
    Description: The connection can be built but the user name or password is not valid.
    Possible values: {300} (the only value)
  - **TAG_FTP_INVALID_FOLDER**
    Constant: jet.cs.util.APIConst.TAG_FTP_INVALID_FOLDER
    Description: The connection can be built but the directory where the report result files reside cannot be found.
    Possible values: {400} (the only value)
- **jrs.ftpRst**
  Constant: jet.cs.util.APIConst.TAG_TO_FTP_RST
  Description: Specifies whether the FTP has RST (JReport result) format.
  Possible values: {true, false}
  - **jrs.ftpRstFn**
    Constant: jet.cs.util.APIConst.TAG_FTP_RST_FILE
    Description: Specifies the file name of the FTP RST result.
    Possible values: FTPrstFileName
  - **jrs.ftpZipRst**
    Constant: jet.cs.util.APIConst.TAG_FTP_ZIP_RESULT
    Description: Specifies whether to zip the FTP RST result.
    Possible values: {true, false}
  - **jrs.ftpRstPcs**
    Constant: jet.cs.util.APIConst.TAG_FTP_RST_PRECISION
    Description: Indicates the precision level of the RST result.
    Possible values: PRECISION_ANY, PRECISION_LOW, PRECISION_HIGH
- **jrs.ftpHtml**
  Constant: jet.cs.util.APIConst.TAG_TO_FTP_HTML
  Description: Specifies whether the FTP has HTML format.
  Possible values: {true, false}
  - **jrs.ftpHtmlFn**
    Constant: jet.cs.util.APIConst.TAG_FTP_HTML_FILE
    Description: Specifies the file name of the FTP HTML format.
    Possible values: ftpHtmlFilename
  - **jrs.ftpHtmNoM**
    Constant: jet.cs.util.APIConst.TAG_FTP_NO_MARGIN_HTML
    Description: Specifies if no margin is set in the HTML page.
    Possible values: {true, false}
  - **jrs.ftpHtmMF**
    Constant: jet.cs.util.APIConst.TAG_FTP_IS_MULTI_FILES
    Description: Specifies whether the HTML result is generated to multiple files or to one file.
    Possible values: {true, false}
    - **jrs.ftpHtmEC**
      Constant: jet.cs.util.APIConst.TAG_FTP_EMBEDDED_CSS
      Description: Specifies whether to embed the cascading style sheet (CSS) in the exported HTML files when generating multiple files. When false, the .css file will be generated individually.
      Possible values: {true, false}
    - **jrs.ftpHtmHL**
      Constant: jet.cs.util.APIConst.TAG_FTP_HAS_HYPERLINK
      Description: Specifies whether the hyperlink is generated.
      Possible values: {true, false}
    - **jrs.ftpHtmPN**
      Constant: jet.cs.util.APIConst.TAG_FTP_HAS_PAGE_NUMBER
      Description: Specifies whether the page number is generated.
      Possible values: {true, false}
  - **jrs.ftpHtmUS**
    Constant: jet.cs.util.APIConst.TAG_FTP_USE_SCOPE
    Description: Specifies whether to use the Scope attribute or not.
    Possible values: {true, false}
  - **jrs.ftpHtmUSO**
    Constant: jet.cs.util.APIConst.TAG_FTP_USE_SECTION508_OUTPUT
    Description: Specifies whether to use Section 508 compliant output or not.
    Possible values: {true, false}
  - **jrs.ftpHtmUHT**
    Constant: jet.cs.util.APIConst.TAG_FTP_USE_HTML_TABLE
    Description: Specifies whether to use HTML data table or not.
    Possible values: {true, false}
  - **jrs.ftpHtmRFS**
    Constant: jet.cs.util.APIConst.TAG_FTP_RELATIVE_FONT_SIZE
    Description: Specifies whether to use relative font size to generate the HTML page.
    Possible values: {true, false}
  - **jrs.ftpHtmRsl**
    Constant: jet.cs.util.APIConst.TAG_FTP_HTML_RESOLUTION
    Description: Specifies the resolution of the HTML page, in dpi.
    Possible values: 96
  - **jrs.ftpHtmWB**
    Constant: jet.cs.util.APIConst.TAG_FTP_WEB_BROWSER
    Description: Specifies the web browser for which the HTML result adapts.
    Possible values: {0, 1}. 0 - IE or Chrome, 1 - Firefox
  - **jrs.ftpHtmDD**
    Constant: jet.cs.util.APIConst.TAG_FTP_DRILLDOWN
    Description: Specifies whether the HTML page can be drilled down.
    Possible values: {true, false}
  - **jrs.ftpHtmFC**
    Constant: jet.cs.util.APIConst.TAG_FTP_FORMAT_CHART
    Description: Specifies the image format in HTML file.
    Possible values: {0, 1, 2, 3}. 0 - Auto-select, 1 - GIF, 2 - JPEG, 3 - PNG
  - **jrs.ftpHtmICA**
    Constant: jet.cs.util.APIConst.TAG_FTP_IS_CHART_APPLET
    Description: Specifies whether the chart in HTML file is an applet.
    Possible values: {true, false}
  - **jrs.ftpHtmTO**
    Constant: jet.cs.util.APIConst.TAG_FTP_TEXT_OVERFLOW
    Description: Specifies the text overflow type.
    Possible values: {0, 1}. 0 - HIDDEN, 1 - VISIBLE
  - **jrs.ftpHtmPcs**
    Constant: jet.cs.util.APIConst.TAG_FTP_HTML_PRECISION
    Description: Indicates the precision level of the HTML result.
    Possible values: {PRECISION_ANY, PRECISION_LOW, PRECISION_HIGH}
- **jrs.ftpPdf**
  Constant: jet.cs.util.APIConst.TAG_TO_FTP_PDF
  Description: Specifies whether the FTP has PDF format.
  Possible values: {true, false}
  - **jrs.ftpPdfFn**
    Constant: jet.cs.util.APIConst.TAG_FTP_PDF_FILE
    Description: Specifies the file name of FTP PDF format.
    Possible values: ftppdffilename
  - **jrs.ftpPdfNoM**
    Constant: jet.cs.util.APIConst.TAG_FTP_NO_MARGIN_PDF
    Description: Specifies if no margin is set in the PDF page.
    Possible values: {true, false}
  - **jrs.ftpPdfIsC**
    Constant: jet.cs.util.APIConst.TAG_FTP_IS_COMPRESS_PDF
    Description: Specifies whether to compress image in PDF file.
    Possible values: {true, false}
    - **jrs.ftpPdfCIm**
      Constant: jet.cs.util.APIConst.TAG_FTP_COMPRESS_IMAGE_PDF
      Description: Specifies the image compress ratio in PDF file. It is measured by percentage.
      Possible values: 20
  - **jrs.ftpPdfDD**
    Constant: jet.cs.util.APIConst.TAG_FTP_DRILLDOWN_PDF
    Description: Specifies whether drilldown is enabled in the PDF page.
    Possible values: {true, false}
  - **jrs.ftpPdfEcr**
    Constant: jet.cs.util.APIConst.TAG_FTP_PDF_ENCRYPT
    Description: Specifies whether PDF encrypt is enable or not.
    Possible values: {true, false}
    - **jrs.ftpPdfCmp**
      Constant: jet.cs.util.APIConst.TAG_FTP_PDF_COMPATIBITLY
      Description: Specifies the PDF compatibility.
      Possible values: {0, 1}. 0 - Acrobat 3.0 and later, 1 - Acrobat 5.0 and later
    - **jrs.ftpPdfPwd**
      Constant: jet.cs.util.APIConst.TAG_FTP_PDF_PWD
      Description: Specifies the PDF password.
      Possible values: 12345678
    - **jrs.hasftpPdfPmP**
      Constant: jet.cs.util.APIConst.TAG_HAS_FTP_PDF_PERMIS_PASW
      Description: Specifies whether to set PDF permission password.
      Possible values: {true, false}
    - **jrs.ftpPdfPmP**
      Constant: jet.cs.util.APIConst.TAG_FTP_PDF_PERMIS_PASW
      Description: Specifies the PDF permission password.
    - **jrs.ftpPdfPrt**
      Constant: jet.cs.util.APIConst.TAG_FTP_PDF_PRINTALLOW
      Description: Specifies the PDF print allow value.
      Possible values: {0, 1, 2}. 0 - none, 1 - low resolution (150 dpi), 2 - high resolution
    - **jrs.ftpPdfChg**
      Constant: jet.cs.util.APIConst.TAG_FTP_PDF_CHANGALLOW
      Description: Specifies the PDF change allow value.
      Possible values: {0, 1, 2, 3, 4}. 0 - none, 1 - inserting, deleting and rotating pages, 2 - filling in form fields and signing, 3 - commenting filling in form fields and signing, 4 - any except extracting pages
    - **jrs.ftpPdfECp**
      Constant: jet.cs.util.APIConst.TAG_FTP_PDF_CHECK_ENABLECOPY
      Description: Specifies whether to be able to copy the PDF page.
      Possible values: {true, false}
    - **jrs.ftpPdfEAc**
      Constant: jet.cs.util.APIConst.TAG_FTP_PDF_CHECK_ENABLEACCESS
      Description: Specifies whether can access the PDF page.
      Possible values: {true, false}
  - **jrs.ftpPdfSgn**
    Constant: jet.cs.util.APIConst.TAG_FTP_PDF_SIGN
    Description: Specifies whether to be able to sign the PDF page.
    Possible values: {true, false}
    - **jrs.ftpPdfSM**
      Constant: jet.cs.util.APIConst.TAG_FTP_PDF_SIGN_METHOD
      Description: Specifies the PDF digital sign method.
      Possible values: {0, 1, 2}. 0 - Windows Certificate Security, 1 - Default Certificate Security, 2 - VeriSign Digital Signatures
    - **jrs.ftpPdfDP**
      Constant: jet.cs.util.APIConst.TAG_FTP_PDF_DIGITAL_PATH
      Description: Specifies the PDF digital sign path.
      Possible values: digitalpath
    - **jrs.ftpPdfPsw**
      Constant: jet.cs.util.APIConst.TAG_FTP_PDF_USER_PASW
      Description: Specifies the password for the digital ID file.
    - **jrs.ftpPdfDR**
      Constant: jet.cs.util.APIConst.TAG_FTP_PDF_DOCUMENT_REASON
      Description: Specifies the reason for signing the document.
      Possible values: SignReason
    - **jrs.ftpPdfEdt**
      Constant: jet.cs.util.APIConst.TAG_FTP_PDF_EDIT_YOURSELF
      Description: Edits the reason for signing the document.
    - **jrs.ftpPdfLoc**
      Constant: jet.cs.util.APIConst.TAG_FTP_PDF_LOCATION
      Description: Specifies the PDF location message.
      Possible values: pdffilelocation
    - **jrs.ftpPdfCnt**
      Constant: jet.cs.util.APIConst.TAG_FTP_PDF_CONTACT_INF
      Description: Specifies the PDF contact information.
      Possible values: contactname
  - **jrs.ftpPdfPM**
    Constant: jet.cs.util.APIConst.TAG_FTP_PRINT_MODE_PDF
    Description: Specifies whether it is for print mode.
    Possible values: {true, false}
  - **jrs.ftpPdfToc**
    Constant: jet.cs.util.APIConst.TAG_FTP_TOC_PDF
    Description: Specifies whether the PDF has TOC.
    Possible values: {true, false}
  - **jrs.ftpPdfGTp**
    Constant: jet.cs.util.APIConst.TAG_FTP_GIF_TRANSPARENT_PDF
    Description: Specifies whether the figure is transparent or not.
    Possible values: {true, false}
  - **jrs.ftpPdfPcs**
    Constant: jet.cs.util.APIConst.TAG_FTP_PDF_PRECISION
    Description: Indicates the precision level of PDF.
    Possible values: {PRECISION_ANY, PRECISION_LOW, PRECISION_HIGH}
- **jrs.ftpExl**
  Constant: jet.cs.util.APIConst.TAG_TO_FTP_EXCEL
  Description: Specifies whether the FTP has Excel format.
  Possible values: {true, false}
  - **jrs.ftpExlFn**
    Constant: jet.cs.util.APIConst.TAG_FTP_EXCEL_FILE
    Description: Specifies the file name of FTP Excel format.
    Possible values: ftpexcelfilename
  - **jrs.ftpExlWW**
    Constant: jet.cs.util.APIConst.TAG_FTP_EXCEL_WORDWRAP
    Description: Specifies whether the Excel has wordwrap format.
    Possible values: {true, false}
  - **jrs.ftpExlLyt**
    Constant: jet.cs.util.APIConst.TAG_FTP_EXCEL_LAYOUT
    Description: Specifies whether to use new Excel layout.
    Possible values: {true, false}
  - **jrs.ftpExl2k**
    Constant: jet.cs.util.APIConst.TAG_FTP_EXCEL2000
    Description: Specifies whether the Excel has Data Format, which means only the report data will be exported without format.
    Possible values: {true, false}
  - **jrs.ftpExlPcs**
    Constant: jet.cs.util.APIConst.TAG_FTP_EXCEL_PRECISION
    Description: Indicates the precision level of Excel.
    Possible values: {PRECISION_ANY, PRECISION_LOW, PRECISION_HIGH}
  - **jrs.ftpExlAdv**
    Constant: jet.cs.util.APIConst.TAG_FTP_EXCEL_ADVANCED
    Description: Specifies whether the advanced Excel setting is enabled or not.
    Possible values: {true, false}
  - **jrs.ftpExlHS**
    Constant: jet.cs.util.APIConst.TAG_FTP_HAS_SHAPES
    Description: Specifies whether to include the drawing objects in the exported Excel file, such as line, oval, and box.
    Possible values: {true, false}
  - **jrs.ftpExlPrH**
    Constant: jet.cs.util.APIConst.TAG_FTP_PRINT_HEADER
    Description: whether to include page header when printing the exported Excel file.
    Possible values: {true, false}
  - **jrs.ftpExlPrF**
    Constant: jet.cs.util.APIConst.TAG_FTP_PRINT_FOOTER
    Description: whether to include page footer when printing the exported Excel file.
    Possible values: {true, false}
  - **jrs.ftpExlPGL**
    Constant: jet.cs.util.APIConst.TAG_FTP_PRINT_GRIDLINES
    Description: whether to include gridlines when printing the exported Excel file.
    Possible values: {true, false}
- **jrs.ftpTxt**
  Constant: jet.cs.util.APIConst.TAG_TO_FTP_TXT
  Description: Specifies whether the FTP has TXT format.
  Possible values: {true, false}
  - **jrs.ftpTxtFn**
    Constant: jet.cs.util.APIConst.TAG_FTP_TEXT_FILE
    Description: Specifies the file name of FTP TXT format.
    Possible values: FtpTextFileName
  - **jrs.ftpTxtCmp**
    Constant: jet.cs.util.APIConst.TAG_FTP_TXT_COMPRESS
    Description: Specifies whether to compress the published text file.
    Possible values: {true, false}
  - **jrs.ftpTxtWin**
    Constant: jet.cs.util.APIConst.TAG_FTP_TXT_WINDOWS
    Description: Specifies whether to apply "\r\n" for OS390 and Windows or "\n" for UNIX.
    Possible values: {true, false}
  - **jrs.ftpTxtHF**
    Constant: jet.cs.util.APIConst.TAG_FTP_HASHEADFOOT
    Description: Specifies whether to generate the page header or footer.
    Possible values: {true, false}
  - **jrs.ftpTxtIDl**
    Constant: jet.cs.util.APIConst.TAG_FTP_TEXT_IS_DELIMITER
    Description: Specifies whether the delimiter is tab or CSV.
    Possible values: {TAG_FTP_IS_CSV, TAG_FTP_IS_TAB}
    - **jrs.ftpTxtCsv**
      Constant: jet.cs.util.APIConst.TAG_FTP_IS_CSV
      Description: Specifies whether the result file is CSV format.
      Possible values: {true, false}
    - **jrs.ftpTxtTab**
      Constant: jet.cs.util.APIConst.TAG_FTP_IS_TAB
      Description: Specifies whether the delimiter is tab char.
      Possible values: {true, false}
  - **jrs.ftpTxtDlm**
    Constant: jet.cs.util.APIConst.TAG_FTP_DELIMITER
    Description: Specifies a delimiter which will be used in delimited format.
    Possible values: "", " "
  - **jrs.ftpTxtNrm**
    Constant: jet.cs.util.APIConst.TAG_FTP_IS_NORMAL_TEXT
    Description: Specifies whether the text file is a delimited format or not.
    Possible values: {true, false}
  - **jrs.ftpTxtPcs**
    Constant: jet.cs.util.APIConst.TAG_FTP_TEXT_PRECISION
    Description: Indicates the precision level of Text.
    Possible values: {RECISION_ANY, PRECISION_LOW, PRECISION_HIGH}
  - **jrs.ftpTxtRes**
    Constant: jet.cs.util.APIConst.TAG_FTP_RESOLUTION
    Description: Specifies whether to set resolution of text file.
    Possible values: {true, false}
  - **jrs.ftpTxtQM**
    Constant: jet.cs.util.APIConst.TAG_FTP_IS_QUOTEMARK
    Description: Specifies to use quote marks in the text file.
    Possible values: {true, false}
  - **jrs.ftpTxtRpt**
    Constant: jet.cs.util.APIConst.TAG_FTP_IS_REPEAT
    Description: Specifies whether the value of the previous cell in the same column will be used when a cell in the exported CSV text has no value.
    Possible values: {true, false}
  - **jrs.ftpTxtChH**
    Constant: jet.cs.util.APIConst.TAG_FTP_CHAR_HEIGHT
    Description: Specifies the user defined character height.
    Possible values: {true, false}
  - **jrs.ftpTxtChW**
    Constant: jet.cs.util.APIConst.TAG_FTP_CHAR_WIDTH
    Description: Specifies the user defined character width.
    Possible values: {true, false}
- **jrs.ftpRtf**
  Constant: jet.cs.util.APIConst.TAG_TO_FTP_RTF
  Description: Specifies whether the FTP has RTF format.
  Possible values: {true, false}
  - **jrs.ftpRtfFn**
    Constant: jet.cs.util.APIConst.TAG_FTP_RTF_FILE
    Description: Specifies the file name of FTP RTF format.
    Possible values: FtpRtfFileName
  - **jrs.ftpRtfNoM**
    Constant: jet.cs.util.APIConst.TAG_FTP_NO_MARGIN_RTF
    Description: Specifies whether there is no margin in the RTF file.
    Possible values: {true, false}
  - **jrs.ftpRtfBE**
    Constant: jet.cs.util.APIConst.TAG_FTP_BEST_EDITING_RTF
    Description: Specifies whether to apply flow layout when exporting the report to RTF.
    Possible values: {true, false}
  - **jrs.ftpRtfPcs**
    Constant: jet.cs.util.APIConst.TAG_FTP_RTF_PRECISION
    Description: indicate the precision level of RTF.
    Possible values: {PRECISION_ANY, PRECISION_LOW, PRECISION_HIGH}
- **jrs.ftpXml**
  Constant: jet.cs.util.APIConst.TAG_TO_FTP_XML
  Description: Specifies whether the FTP has XML format.
  Possible values: {true, false}
  - **jrs.ftpXmlFn**
    Constant: jet.cs.util.APIConst.TAG_FTP_XML_FILE
    Description: Specifies the file name of FTP XML format.
    Possible values: FtpXmlFileName
  - **jrs.ftpXmlIOD**
    Constant: jet.cs.util.APIConst.TAG_FTP_IS_ONLY_DATA
    Description: If true, the exported XML file will only contain the database column information. The exported XML schema file will only contain the structure information of the report. If false, the exported XML file will also contain elements controlled by formulas, and the exported XML schema file will contain all the detailed information from the report, including all the property values of each report object.
    Possible values: {true, false}
  - **jrs.ftpXmlXsd**
    Constant: jet.cs.util.APIConst.TAG_FTP_XSDFILE
    Description: Specifies the directory and the name of an existing XML schema (.xsd) file.
    Possible values: FtpXsdFileName
  - **jrs.ftpXmlPcs**
    Constant: jet.cs.util.APIConst.TAG_FTP_XML_PRECISION
    Description: Indicates the precision level of XML.
    Possible values: {PRECISION_ANY, PRECISION_LOW, PRECISION_HIGH}
- **jrs.ftpPs**
  Constant: jet.cs.util.APIConst.TAG_TO_FTP_PS
  Description: Specifies whether the FTP has PostScript format.
  Possible values: {true, false}
  - **jrs.ftpPsFn**
    Constant: jet.cs.util.APIConst.TAG_FTP_PS_FILE
    Description: Specifies the file name of FTP PostScript format.
    Possible values: FtpPsfilename
  - **jrs.ftpPsNoM**
    Constant: jet.cs.util.APIConst.TAG_FTP_NO_MARGIN_PS
    Description: Specifies whether there is no margin in the PostScript file.
    Possible values: {true, false}
  - **jrs.ftpPsPcs**
    Constant: jet.cs.util.APIConst.TAG_FTP_PS_PRECISION
    Description: Indicates the precision level of PostScript.
    Possible values: {PRECISION_ANY, PRECISION_LOW, PRECISION_HIGH}
- **jrs.ftp_end**
  Constant: jet.cs.util.APIConst.TAG_FTP_END
  Description: Specifies the JReport FTP delimiter for multiple FTPs.
- **jrs.ftpIsDht**
  Constant: jet.cs.util.APIConst.TAG_FTP_IS_DHTML
  Description: Specifies whether to show Page Report Studio page.
  Possible values: {true, false}

Time condition properties

When scheduling reports via URL, you can specify time condition for when the reports will run using the following properties:

- **jrs.launch_type**
  Constant: jet.cs.util.APIConst.TAG_LAUNCH_TYPE
  Description: Specifies to run a report immediately, at a specific time or periodically.
  Possible values: {0, 1, 8}. 0 - run immediately (APIConst.IMMEDIATE), 1 - run at a specified time (APIConst.AT_TIME), 8 - run periodically (APIConst.PERIODICALLY). When jrs.launch_type=1, refer to the section Properties for a specific date and time to define a specific time. When jrs.launch_type=8, refer to the Properties for periodical time section to define the periodical time.

- **jrs.timezone**
  Constant: jet.cs.util.APIConst.TAG_TIMEZONE
  Description: Specifies the timezone.
  Possible values: timezone ID strings of java default is current locale.

Properties for a specific date and time (for jrs.launch_type=1)

- **jrs.exe_date**
  Constant: jet.cs.util.APIConst.TAG_EXE_DATE
  Description: Specifies a date.

- **jrs.exe_year**
  Constant: jet.cs.util.APIConst.TAG_EXE_YEAR
  Description: Specifies a year of 4 numbers.
  Possible values: 1980 to 2099

- **jrs.exe_month**
  Constant: jet.cs.util.APIConst.TAG_EXE_MONTH
  Description: Specifies a month, for example, May.
  Possible values: 0 to 11. 0 is January, 1 is Febuary, ..., and 11 is December.

- **jrs.exe_day**
  Constant: jet.cs.util.APIConst.TAG_EXE_DAY
  Description: Specifies a day number, for example, the 3rd day.
  Possible values: 1 to 31

- **jrs.exe_hour**
  Constant: jet.cs.util.APIConst.TAG_EXE_HOUR
  Description: Specifies an hour number (when jrs.is_hourly is false), for example, 10 o'clock.
  Possible values: 0 to 23

- **jrs.exe_min**
  Constant: jet.cs.util.APIConst.TAG_EXE_MIN
  Description: Specifies a minute number (when jrs.is_hourly is false), for example, 10:**15** am.
  Possible values: 0 to 59

- **jrs.exe_sec**
  Constant: jet.cs.util.APIConst.TAG_EXE_SEC
  Description: Specifies a second number (when jrs.is_hourly is false), for example, 10:15:**40** am.
  Possible values: 0 to 59

Properties for periodical time (for jrs.launch_type=8)

- **jrs.is_after**
  Constant: jet.cs.util.APIConst.SCH_IS_AFTER
  Description: Specifies to run after the specified date and time, also the start time of the whole periodical time. If the value is true, refer to the section Properties for a specific time to define the specific time.
  Possible values: {true, false}

- **jrs.is_before**
  Constant: jet.cs.util.APIConst.SCH_IS_BEFORE
  Description: Specifies to run before the specified date and time. If the value is true, the specified time is represented by the jrs.cease_* properties as follows.
  Possible values: {true, false}

  - **jrs.cease_year**
    Constant: jet.cs.util.APIConst.TAG_CEASE_YEAR
    Description: Specifies a year of 4 numbers.
    Possible values: 1980 to 2099

  - **jrs.cease_month**
    Constant: jet.cs.util.APIConst.TAG_CEASE_MONTH
    Description: Specifies a month, for example, May.
    Possible values: 0 to 11. 0 is January, 1 is Febuary, ..., and 11 is December.

  - **jrs.cease_day**
    Constant: jet.cs.util.APIConst.TAG_CEASE_DAY
    Description: Specifies a day number, for example, the May **3rd**, 2012.
    Possible values: 1 to 31

  - **jrs.cease_hour**
    Constant: jet.cs.util.APIConst.TAG_CEASE_HOUR
    Description: Specifies an hour number (when jrs.is_hourly is false), for example, 10 o'clock.
    Possible values: 0 to 23

  - **jrs.cease_min**
    Constant: jet.cs.util.APIConst.TAG_CEASE_MIN
    Description: Specifies a minute number (when jrs.is_hourly is false), for example, 10:**15** am.
    Possible values: 0 to 59

  - **jrs.cease_sec**
    Constant: jet.cs.util.APIConst.TAG_CEASE_SEC
    Description: Specifies a second number (when jrs.is_hourly is false), for example, 10:15:**40** am.
    Possible values: 0 to 59

- **jrs.days_id**
  Constant: jet.cs.util.APIConst.SCH_DAYS_ID
  Description: Specifies whether the schedule will run daily, weekly or monthly.
  Possible values: {0, 1, 2}. 0 - daily, 1 - weekly, 2 - monthly

Properties for daily (for jrs.days_id=0)

- **jrs.is_weekday**
  Constant: jet.cs.util.APIConst.SCH_IS_WEEKDAY
  Description: Specifies whether to run a report on each weekday (Monday to Friday).
  Possible values: {true, false}. The default is false.

- **jrs.day**
  Constant: jet.cs.util.APIConst.SCH_DAY
  Description: Specifies that the report will run once every how many days specified here (when jrs.is_weekday is false). For example, if the value is set to 1, the report will run once everyday. If the value is set to 2, the report will run once every 2 days.
  Possible values: 1 to 999 days

Properties for weekly (for jrs.days_id=1)

- **jrs.week**
  Constant: jet.cs.util.APIConst.SCH_WEEK
  Description: Specifies that the report will run once every how many weeks specified here.
  Possible values: 1 to 99 weeks

- **jrs.weekdays**
  Constant: jet.cs.util.APIConst.SCH_WEEKDAYS
  Description: Specifies on which days to run in a week.
  Possible values: A digit string. {0, 1, 2, 3, 4, 5, 6}. 0 - Sunday, 1 - Monday, 2 - Tuesday, 3 - Wednesday, 4 - Thursday, 5 - Friday, 6 - Saturday
  Example: jrs.weekdays=05 means running on Sunday and Friday.

Properties for monthly (for jrs.days_id=2)

- **jrs.is_day**
  Constant: jet.cs.util.APIConst.SCH_IS_DAY
  Description: Specifies whether to run on the xth day in a month.
  Possible values: {true, false}. The default is true.

- **jrs.day**
  Constant: jet.cs.util.APIConst.SCH_DAY
  Description: Specifies to run on the xth day in a month (when jrs.is_day is true).
  Possible values: 1th to 31th. The default is 1.

- **jrs.week**
  Constant: jet.cs.util.APIConst.SCH_WEEK
  Description: Specifies to run on the xth week in a month (when jrs.is_day is false).
  Possible values: {0, 1, 2, 3, 4}. 0 - first, 1 - second, 2 - third, 3 - fourth, 4 - last week. The default is 0.

- **jrs.weekday**
  Constant: jet.cs.util.APIConst.SCH_WEEKDAY
  Description: Specifies to run on which day in the week (jrs.is_day is false).
  Possible values: {1, 2, 3, 4, 5, 6, 7}. 1 - Sunday, 2 - Monday, 3 - Tuesday, 4 - Wednesday, 5 - Thursday, 6 - Friday, 7 - Saturday. The default is 7.

- **jrs.month**
  Constant: jet.cs.util.APIConst.SCH_MONTH
  Description: Specifies that the report will run once every how many months specified here.
  Possible values: 1 to 6 months. The default is 1.

Properties for a specific time

- **jrs.hour**
  Constant: jet.cs.util.APIConst.SCH_HOUR
  Description: Specifies to run at which hour in a day.
  Possible values: 1 to 12. The default is 1.

- **jrs.min**
  Constant: jet.cs.util.APIConst.SCH_MIN
  Description: Specifies to run at which minute in a hour.
  Possible values: 0 to 59. The default is 0.

- **jrs.is_pm**
  Constant: jet.cs.util.APIConst.SCH_IS_PM
  Description: Specifies whether the time is PM or AM in a day.
  Possible values: {true, false}. The default is true.

Properties for hourly

- **jrs.is_hourly**
  Constant: jet.cs.util.APIConst.SCH_IS_HOURLY
  Description: Specifies to run the schedule every hour.
  Possible values: {true, false}. The default is true.

- **jrs.hours**
  Constant: jet.cs.util.APIConst.SCH_HOURS
  Description: Specifies that the report will run once every how many hours specified here.
  Possible values: 1 to 99 hours. The default is 1.

- **jrs.at_min**
  Constant: jet.cs.util.APIConst.SCH_AT_MIN
  Description: Specifies to run at which minute in an hour.
  Possible values: 0 to 59. The default is 0.

- **jrs.is_between**
  Constant: jet.cs.util.APIConst.SCH_IS_BETWEEN
  Description: Specifies whether to run between the start time and the finish time. When the value is true, refer to the section Properties for when jrs.is_between=true to define a time duration.
  Possible values: {true, false}

Properties for minutely

- **jrs.is_minutely**
  Constant: jet.cs.util.APIConst.SCH_IS_MINUTELY
  Description: Specifies whether to run the schedule every minute.
  Possible values: {true, false}

  Note: The property jrs.is_hourly has the higher priority. That is, if you set both jrs.is_hourly and jrs.is_minutely to true, the schedule will run the task based on hour.

- **jrs.minutes**
  Constant: jet.cs.util.APIConst.SCH_MINUTES
  Description: Specifies that the report will run once every how many minutes specified here.
  Possible values: 1 to 999 minutes

- **jrs.is_between**
  Constant: jet.cs.util.APIConst.SCH_IS_BETWEEN
  Description: Specifies whether to run between the start time and the finish time. When the value is true, refer to the section Properties for when jrs.is_between=true to define a time duration.
  Possible values: {true, false}

Properties for when jrs.is_between=true

- **jrs.hour**
  Constant: jet.cs.util.APIConst.SCH_HOUR
  Description: Specifies the start hour of a period of time.
  Possible values: 1-12hour

- **jrs.min**
  Constant: jet.cs.util.APIConst.SCH_MIN
  Description: Specifies the start minute of a period of time.
  Possible values: 0-59minute

- **jrs.is_pm**
  Constant: jet.cs.util.APIConst.SCH_IS_PM
  Description: Specifies whether the start time is PM or AM.
  Possible values: {true, false}

- **jrs.hour2**
  Constant: jet.cs.util.APIConst.SCH_HOUR2
  Description: Specifies the finish hour of a period of time.
  Possible values: 1-12hour. The default is 5.

- **jrs.min2**
  Constant: jet.cs.util.APIConst.SCH_MIN2
  Description: Specifies the finish minute of a period of time.
  Possible values: 0-59minute. The default is 0.

- **jrs.is_pm2**
  Constant: jet.cs.util.APIConst.SCH_IS_PM2
  Description: Specifies whether the finish time is PM or AM.
  Possible values: {true, false}

Notification properties

When scheduling reports via URL, the notification properties are as follows:

- **jrs.notification_emails**
  Constant: jet.cs.util.APIConst.TAG_NOTIFICATION_EMAILS
  Description: Specifies the e-mail notification list for successful/failed scheduled tasks.

- **jrs.success_notify**
  Constant: jet.cs.util.APIConst.TAG_SUCCESS_NOTIFY
  Description: Specifies to send e-mail notification for successful reports.
  Possible values: {true, false}. The default is false.

- **jrs.fail_notify**
  Constant: jet.cs.util.APIConst.TAG_FAIL_NOTIFY
  Description: Specifies to send e-mail notification for failed reports.
  Possible values: {true, false}. The default is false.

Style group properties

When scheduling reports via URL, the style group properties are as follows:

- **jrs.has_style**
  Constant: jet.cs.util.APIConst.TAG_HAS_STYLE
  Description: Specifies to enable the style group that has been set for the report.
  Possible values: {true, false}. The default is false.

- **jrs.style_group**
  Constant: jet.cs.util.APIConst.TAG_STYLE_GROUP
  Description: Specifies the style group which could be StyleName<CSS> or StyleName.css. StyleName<CSS> is CSS Style Sheet whose name is StyleName in the JReport resources system. It may come from a .css file on disk or be created from the CSS editor. StyleName.css is a *.css file on disk which may not follow the CSS style rule and therefore not work. It is recommended to use jrs.style_group=StyleName<CSS> in case StyleName.css fails though both can work most of the time.
  Examples:

  ```
  http://localhost:8888/jinfonet/runReport.jsp?jrs.cmd=jrs.try_vw&jrs.report=/SampleReports/Banded_Link.cls&jrs.catalog=/SampleReports/SampleReports.cat&jrs.path=/SampleReports/Banded_Link.cls&jrs.result_type=8&jrs.auth_uid=admin&jrs.auth_pwd=admin&jrs.has_style=true&jrs.style_group=Warm<CSS>
  ```

  ```
  http://localhost:8888/jinfonet/runReport.jsp?jrs.cmd=jrs.try_vw&jrs.report=/SampleReports/Banded_Link.cls&jrs.catalog=/SampleReports/SampleReports.cat&jrs.path=/SampleReports/Banded_Link.cls&jrs.result_type=8&jrs.auth_uid=admin&jrs.auth_pwd=admin&jrs.has_style=true&jrs.style_group=Warm.css
  ```

Encrypting PDF properties

The following are properties to encrypt the PDF format file:

- **jrs.pdf_encrypt**
  Constant: jet.cs.util.APIConst.TAG_PDF_ENCRYPT
  Description: Specifies whether to enable PDF encryption.
  Possible values: {true, false}

- **jrs.pdf_compatibitly**
  Constant: jet.cs.util.APIConst.TAG_PDF_COMPATIBITLY
  Description: Specifies the encryption type to encrypt a PDF document.
  Possible values: {0, 1}. 0 - Acrobat 3.0 and later, 1 - Acrobat 5.0 and later.

- **jrs.password_pdf**
  Constant: jet.cs.util.APIConst.TAG_PASSWORD_PDF
  Description: Specifies the password to open the PDF document.

- **jrs.has_pdf_permis_pasw**
  Constant: jet.cs.util.APIConst.TAG_HAS_PDF_PERMIS_PASW
  Description: Specifies whether to use a permission password to prevent others from printing and editing the document.
  Possible values: {true, false}

- **jrs.pdf_permis_pasw**
  Constant: jet.cs.util.APIConst.TAG_PDF_PERMIS_PASW
  Description: Specifies the permission password.

- **jrs.pdf_printallow**
  Constant: jet.cs.util.APIConst.TAG_PDF_PRINTALLOW
  Description: Specifies the printing quality for the PDF document.
  Possible values: {0, 1, 2}. 0 - none, 1 - low resolution (150 dpi), 2 - high resolution.

- **jrs.pdf_changallow**
  Constant: jet.cs.util.APIConst.TAG_PDF_CHANGALLOW
  Description: Specifies the editing actions that are permitted in the PDF file.
  Possible values: {0, 1, 2, 3, 4}. 0 - none. 1 - inserting, deleting and rotating pages, 2 - filling in form fields and signing, 3 - commenting filling in form fields and signing, 4 - any except extracting pages.

- **jrs.pdf_enablecopy**
  Constant: jet.cs.util.APIConst.TAG_PDF_CHECK_ENABLECOPY
  Description: Specifies whether to enable selecting and copying the contents of the PDF document.
  Possible values: {true, false}

- **jrs.pdf_enableaccess**
  Constant: jet.cs.util.APIConst.TAG_PDF_CHECK_ENABLEACCESS
  Description: Specifies whether to allow visually impaired users to read the PDF document with window readers. This property is available only if the encryption type is set to Acrobat 5.0 or later Acrobat version.
  Possible values: {true, false}

Appendix 8: Page Report Studio tag library

Page Report Studio provides its tag library, which includes custom tags such as mainpage, head, toc and report. These custom tags can be used to componentize JSP pages, increasing productivity and encapsulating recurring tasks so that they can be reused across more than one application.

For technique topics on the JSP tag library, references can be found in JSP online documentation on the Sun Microsystems website http://www.sun.com.

Page Report Studio tags are classified as follows:

- Embedded tags
- Component tags
- Action tags

Before you can use the Page Report Studio tag library in JReport Server, you should firstly copy the file Jinfonet_DHTML_taglib.tld in `<install_root>\public_html\dhtmljsp` to the folder where your JSP pages are located.

And if you want to use the Page Report Studio tag library in any web server, you should first:

1. Add JRWebDesign.jar in `<install_root>\lib` to the war file you want to publish to the web server.
2. Add Jinfonet_DHTML_taglib.tld in `<install_root>\public_html\dhtmljsp` to the war file in which your JSP pages reside, or alternatively, you can use web.xml to specify a path. For example,

```
<taglib>
    <taglib-uri>Jinfonet_DHTML_taglib</taglib-uri>
    <taglib-location>/WEB-INF/Jinfonet_DHTML_taglib.tld</taglib-location>
</taglib>
```

Embedded tags

dhtml

The dhtml tag is a container for the other tags, and it checks user information, runs a report, and imports resources for the other tags. All tags to be used should be included in this tag. It contains the following attributes:

Attribute	Description
id - string	Specifies the ID of the tag.
tagsetid - string	Specifies the ID of the tag set. This attribute is required.
user - string	The user name for logging in JReport Server.
password - string	The user password for logging in JReport Server.
report - string	The name of the report the user wants to run. This attribute is required.
catalog - string	The name of the catalog for the report. This attribute is required.
report_path - string	The path of the report.
report_params - string	All parameters (name/value pair) that are required for running the report, and the parameters are separated by & character.
catalog _version - integer	The catalog version. Default value is 1.

mainpage

The mainpage tag displays the Page Report Studio page in an IFrame. It contains the following attributes:

Attribute	Description
width=length	The width of the frame. It can be a relative or absolute value, for example, 70% or 1000px.
height=length	The height of the frame. It can be a relative or absolute value, for example, 70% or 1000px.
top=length	The top position of the frame, measured in pixels.
left=length	The left position of the frame, measured in pixels.
align=[top/middle/|bottom/left/right]	Specifies where the frame appears in the web browser window. • **top**: Aligns the frame to the top of the web browser window. • **middle**: Aligns the frame to the middle of the web browser window. • **bottom**: Aligns the frame to the bottom of the web browser window. • **left**: Aligns the frame to the left of the web browser window. • **right**: Aligns the frame to the right of the web browser window.
frameborder=[1/0]	Sets or retrieves whether to display the frame border. Use 1 to show the frame border, or 0 to hide it.
scrolling=[yes/no/auto]	Sets or retrieves whether the frame can be scrolled. • **auto**: Default. The web browser determines whether scrollbars are necessary. • **yes**: The frame can be scrolled. • **no**: The frame cannot be scrolled.
id - string	Specifies the ID of the frame.
name - string	Specifies the frame name.
position=[static/absolute/relative]	Sets or retrieves the type of positioning used for the object . • **static**: Default. Object has no special positioning; it follows the layout rules of HTML. • **absolute**: Object is positioned relative to parent element's position—or to the body object if its parent element is not positioned—using the top and left properties. • **relative**: Object is positioned according to the normal flow, and then offset by the top and left properties.
controls=[userInfoBar|rptSetBar|toolbar|Toc|Dso|report|toolbox|linkbar|all]	All components that the user wants to display. The user can set any one or more of them, or use all to display all the components.

Component tags

toc

The toc tag is for displaying the TOC Browser.

toolbar

The toolbar tag is for displaying the toolbar.

resview

The resview tag is for displaying the Resource View panel.

toolbox

The toolbox tag is for displaying the Toolbox.

button

The button tag is for displaying buttons which match toolbar buttons but they can be placed anywhere in Page Report Studio.

Attribute	Description
buttonid	Identifies the specific button. You can obtain a constant Java class jet.web.dhtml. DHTMLConstant from the package JRWebDesign.jar located at `<install_root>\lib`. In this class, those constants with a prefix "TOOLBAR_" and of int data type represent the buttonid, and you can then know the buttonid according to the constant name. For example, `<jinfonet:button buttonid="<%=String.valueOf(DHTMLConstant.TOOLBAR_NEW)%>">` will display the New button. This attribute is required.

toolboxbutton

The toolboxbutton tag is for displaying buttons which match Toolbox buttons but they can be placed anywhere in Page Report Studio.

Attribute	Description
component_type	Identifies the specific button. These constants are available for component_type: "Label", "Image", "Banded Object", "Table", "Crosstab", and "Chart". For example, `<jinfonet:toolboxbutton component_type="Banded Object"/>` will display the Banded Object button. This attribute is required.

report

The report tag is for Page Report Studio browser.

Attribute	Description
width=length	The width of the report browser, measured in pixels. Use -1 to set it to 100% of Internet Explorer window's width. In other web browsers, -1 means 650 pixels.
height=length	The height of the report browser, measured in pixels. Use -1 to set it to 100% of Internet Explorer window's height. In other web browsers, -1 means 600 pixels. This attribute is required.
top=length	The top position of the report browser, measured in pixels.
left=length	The left position of the report browser, measured in pixels.
id	Specifies the ID of the tag.
class_name	Retrieves the class of the object.
style	Sets a style for the object.
controls=[rptSetBar|linkbar|all]	Specifies whether to display the components. The user can set any one or more of them, or use all to display all the components.

Action tags

sortform

The sortform tag will generate a form for sorting the data. It contains the following attributes:

Attribute	Description
id - string	Specifies the ID of the tag. This attribute is required.
method - string	Sets or retrieves how to send the form data to the server. • **get:** Appends the arguments to the Form Action URL and opens it as if it were an anchor. • **post:** Sends the data through an HTTP post transaction.
forward - string	Specifies the page to redirect. If it starts with /, the root path refers to `<install_root>\public_html`; if with no /, the root refers to `<install_root>\public_html\dhtmljsp`.
target - string	Sets the window or frame at which to target content.
applyto - string	Specifies the instance name of the component on which the sorting will be based. This attribute is required.

filterform

The filterform tag will generate a form for filtering the data. It contains the following attributes:

Attribute	Description
id - string	Specifies the ID of the tag. This attribute is required.
method - string	Sets or retrieves how to send the form data to the server. • **get:** Appends the arguments to the Form Action URL and opens it as if it were an anchor. • **post:** Sends the data through an HTTP post transaction.
forward - string	Specifies the page to redirect. If it starts with /, the root path refers to `<install_root>\public_html`; if with no /, the root refers to `<install_root>\public_html\dhtmljsp`.
target - string	Sets the window or frame at which to target content.
applyto - string	Specifies the instance name of the component on which the sorting will be based. This attribute is required.

column

The column tag will generate a list to provide sorting manners for the sortform tag or field values for the filterform tag. It contains the following attributes:

Attribute	Description
id - string	Specifies the ID of the tag. This attribute is required.
bind_column - string	Specifies the column by which to sort or filter. This attribute is required.
logic - string	Specifies the relationship between the filter criterion and the next one. Its value can be "AND" or "OR". Only for filterform.
maxsize - integer	Specifies the maximum number of the items in the list, only for filterform.
parent_id - string	Specifies the ID of the sortform or filterform tag which includes the column tag. This attribute is required.
class_name - string	Specifies the class name.
style - string	Specifies the style of the list.
width - string	Specifies the width of the list.
top - integer	Specifies the top position of the list.
left - integer	Specifies the left position of the list.

Reference: For examples about the tags, refer to the file ExampleCodeForTags.html in `<install_root>\help\samples\URLSamples`, where some sample code for different tags is provided. To test the examples, save the code to tagtest.jsp and then put it in `<install_root>\public_html\dhtmljsp`.