



Getting Started with JReport

[JReport Product Overview](#)

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

[System Requirements](#)

Both JReport Designer and JReport Server have some basic installation requirements. Before installing them, check your system to make sure all the requirement have been met.

[JReport Licenses](#)

JReport has several special licenses which allow you to use some specific features in JReport.

[Supported Report Databases](#)

JReport supports all of the current mainstream databases as well as most databases which support ODBC or JDBC drivers.

[Test Server Installation](#)

Before you log onto the JReport Server, you have to first start the server.

[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.

[View Sample Reports](#)

After successfully logging onto the JReport Console page, you can then perform tasks according to your requirements, such as viewing reports in different formats, running them via URLs, customizing them by setting properties, etc.

[Create Library Components and Dashboards](#)

You can create dashboards using JDashboard and build a user portal user interface. However, before you can create dashboards, the report developer should first create library components to present data using JReport Designer and then publish them to JReport Server.

[Create Ad Hoc Reports](#)

In Page Report Studio, you can create ad hoc reports based on predefined business views that are published to JReport Server. However, you will only see the New Report link in the console when the current folder has a catalog in it that contains a business view.

[Start a Visual Analysis session](#)

You can visualize the result of every step of business data analysis by dragging and dropping business data fields to the layout module using Visual Analysis.

[Open Sample Report Templates](#)

There are many sample reports in JReport Designer, you can open any one sample report and preview the report result.

[Start Creating New Reports](#)

Before you create new reports using JReport Designer, you have to make it have access to your data source. Currently, seven methods are provided for you to achieve this goal: via JDBC/ODBC Connection, XML Connection, Web Service Connection, MongoDB Connection, HIVE Connection, Hierarchical Data Source, or by creating a User Defined Data Source.

[Build Business Views](#)

To make use of business views in Page Report Studio and Web Report Studio, you need to first define them at report design time in JReport Designer.

[Online Help](#)

Online documentation includes all information about the JReport products.

Note: If you are using Google Chrome, because of its security policy to not allow local file access by default, this guide will not be displayed normally. To solve this, you have to add the parameter --allow-file-access-from-files to the Target property of your Google Chrome shortcut, then launch Chrome with this shortcut and open the guide in it. The guide is located in your <install_root>\help\start folder.





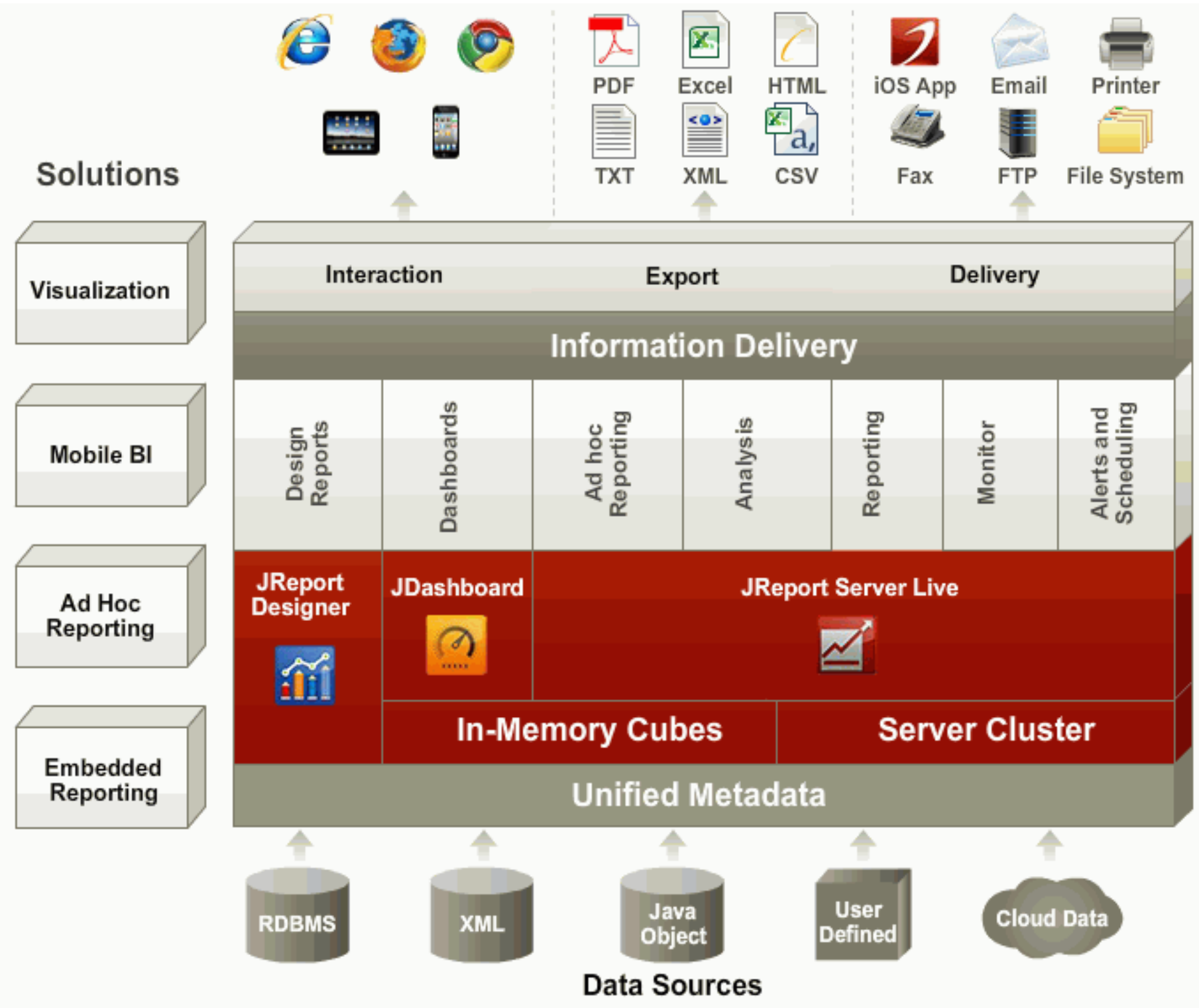
JReport Product Overview

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, etc. 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.





System Requirements for JReport Designer

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

JReport Designer System Requirements		
	Recommended Requirements	Minimum Requirements
OS:	Windows x64, Unix x64 and Linux x64	Windows, Unix, Linux
CPU:	Quad Core processor	Dual Core processor
Free Memory:	4 GB	2 GB
Free Disk:	2 GB	1 GB
JDK:	6 or above	6 or above
Browser:	Latest releases	IE 9, Firefox 20, Chrome 23, Edge

System Requirements for JReport Server

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	IE 9, Firefox 20, Chrome 23, Edge

Both JReport Designer and JReport Server require that a Java Development Kit (JDK) 6 or higher be already installed on your system. You can 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 Designer and 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 some 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 web 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 (the browser for opening a page report) 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

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. 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
MySQL	5.5.24 (64bit)	mysql-connector-java-5.1.125-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 etc)	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

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.





Test Server Installation

Before you log onto the JReport Server, you have to first start the server. This topic shows you how to access JReport Server through a web browser such as Internet Explorer, Chrome, or Firefox.

Starting and logging onto JReport Server

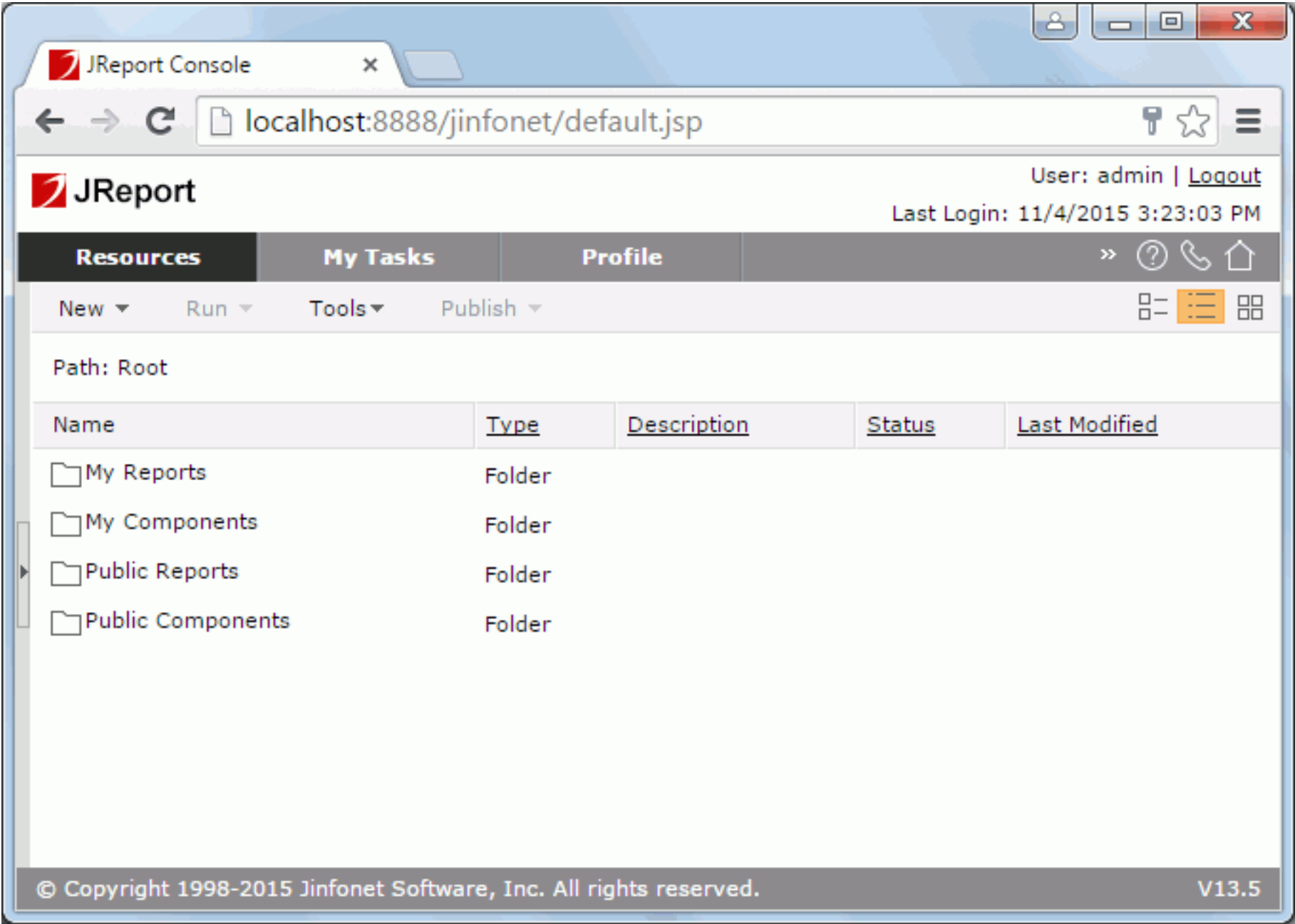
To log onto JReport Server, first start the server via one of the following ways:

- 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 MS-DOS command prompt. For example, assume that JReport Server has been installed in c:\JReport\Server, you can type the following commands:

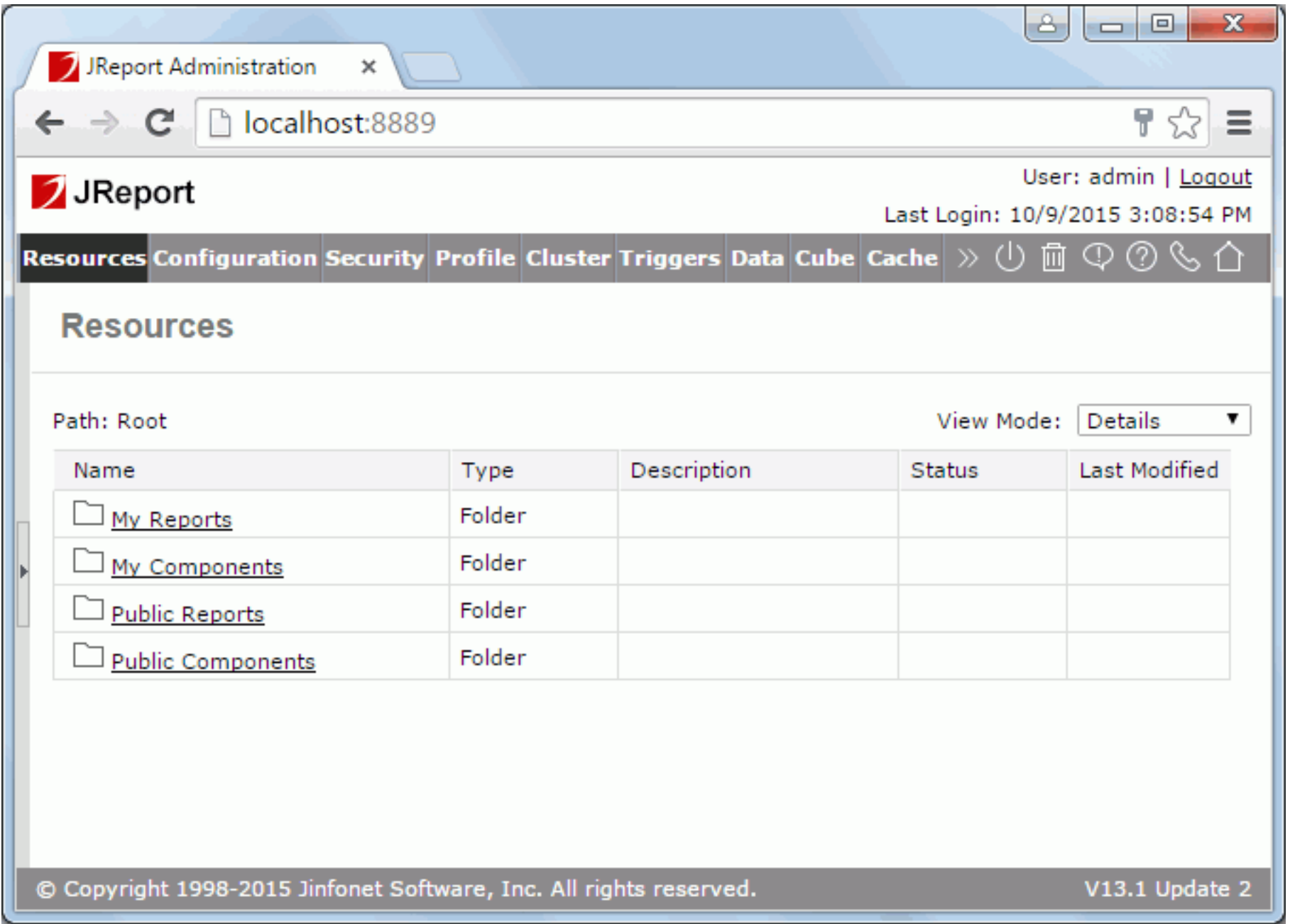
```
C:\>cd JReport\Server\bin
C:\JReport\Server\bin>JRServer.bat
```

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 is admin for both user name and password.
 3. Click the **Login** button. 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 is admin for both user name and password.
 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 your web browser, you can use *localhost* instead of the IP address. You can also open a console window on the server machine and type hostname, then the name of the host will be displayed.



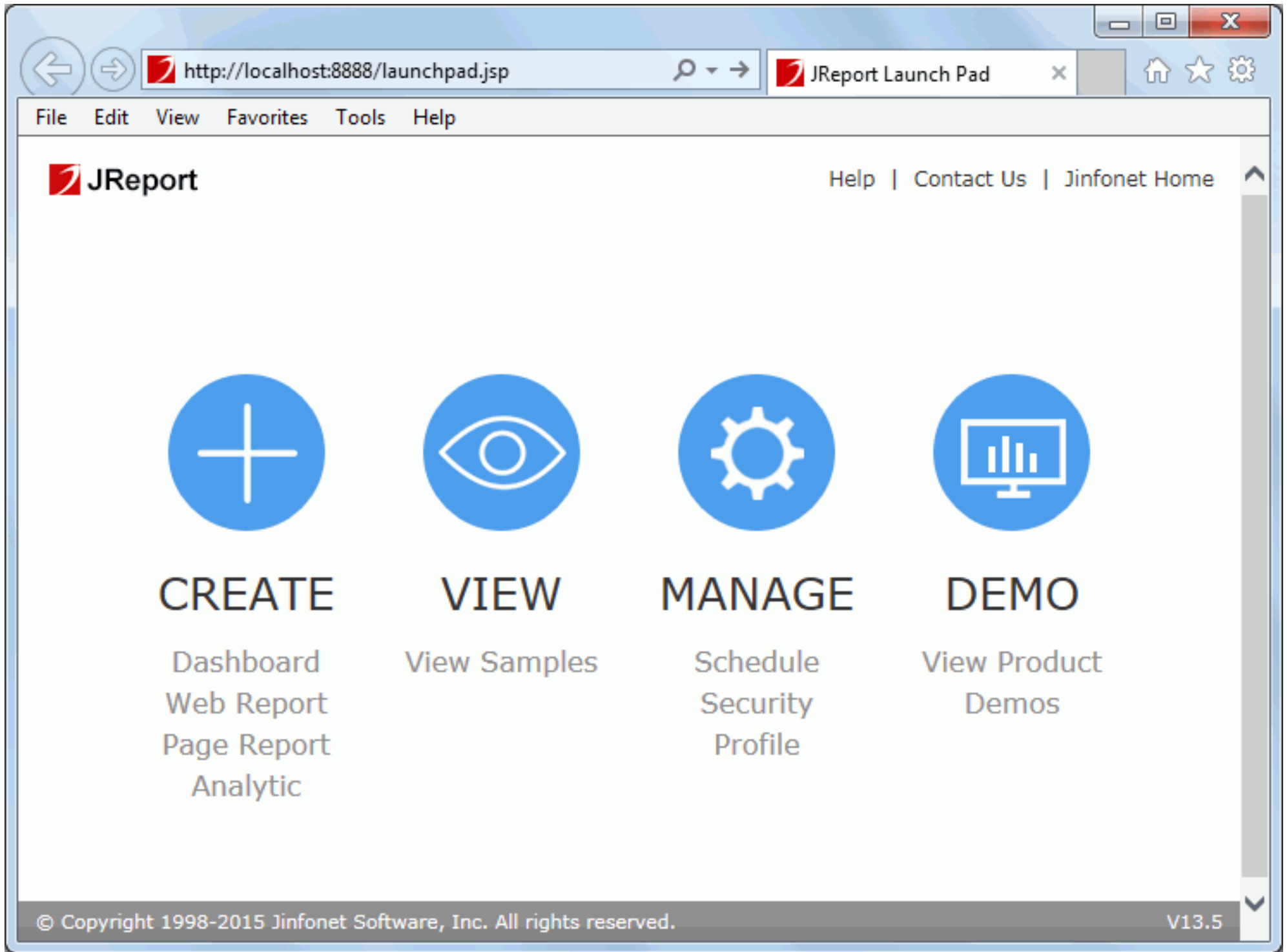


Fast Launch Pad for Local Users

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 JSP's, 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**.




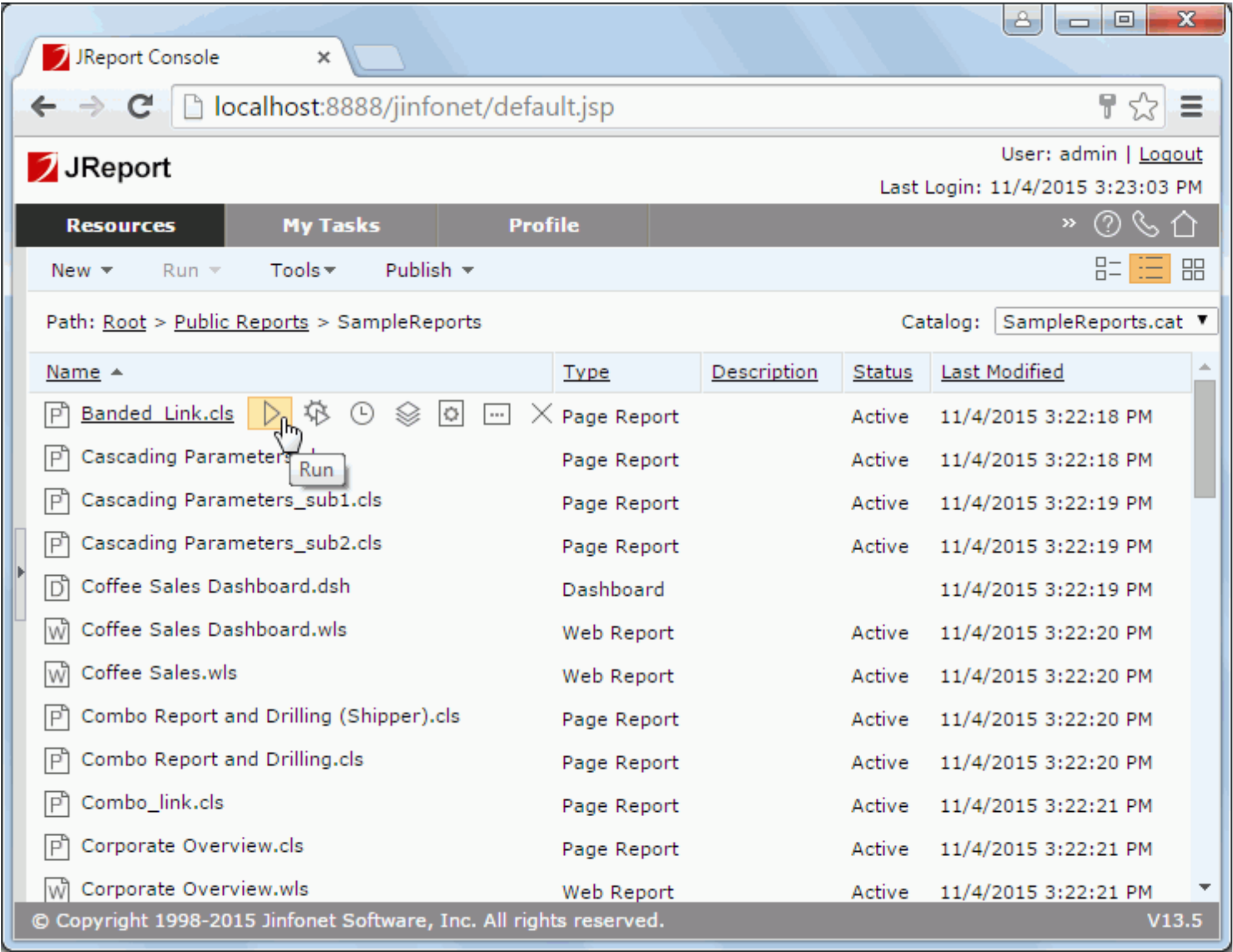


View Sample Reports

After successfully logging onto the JReport Console page, you can then perform tasks according to your requirements, such as viewing reports in different formats, running them via URLs, customizing them by setting properties, etc.

To view a specified 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 row the report is in, then 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, the last-time focused report tab in the page report when it was saved in JReport Designer or the web report will be displayed according to its Default Format for Viewing Report property setting in JReport Designer. If this property of the report tab or the web report is set to Server Setting, the viewing format will be determined by the Default Format for Viewing Report option on the JReport Administration/Console page > Profile > Customize Server Preferences > General tab. However, for page report, if the viewing format of the last-time focused report tab is Page Report, no matter what viewing formats the other report tabs in this report are defined as in JReport Designer, all the report tabs will be opened in Page Report format when you view this report on JReport Server.

If you want to select a specific export format and exactly which report tab in a page report to run, use Advanced Run from the floating menu, context menu or Run menu.





Create Library Components and Dashboards

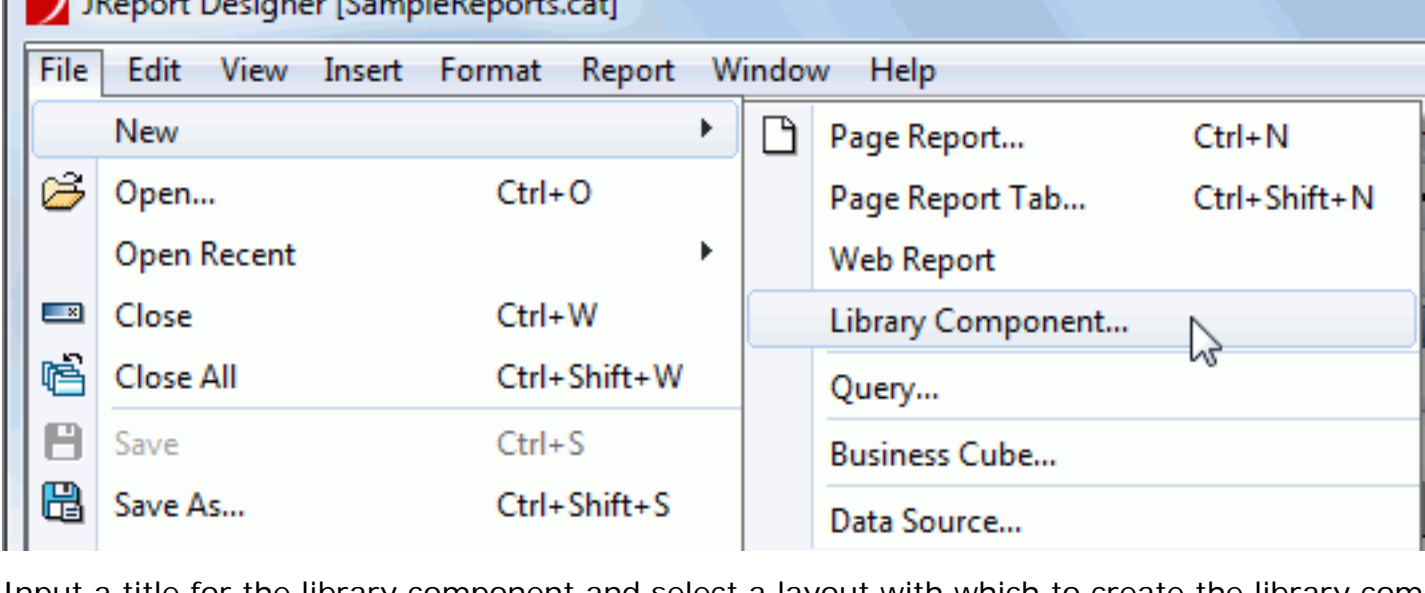
JDashBoard is a new way of information delivery, using a user portal user interface rather than a web report or page report. Users can create, edit and browse dashboards from the JReport Console using JDashBoard.

Library components are the basic members in dashboards for presenting data via intuitive components such as charts, crosstabs, tables, and Google 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.

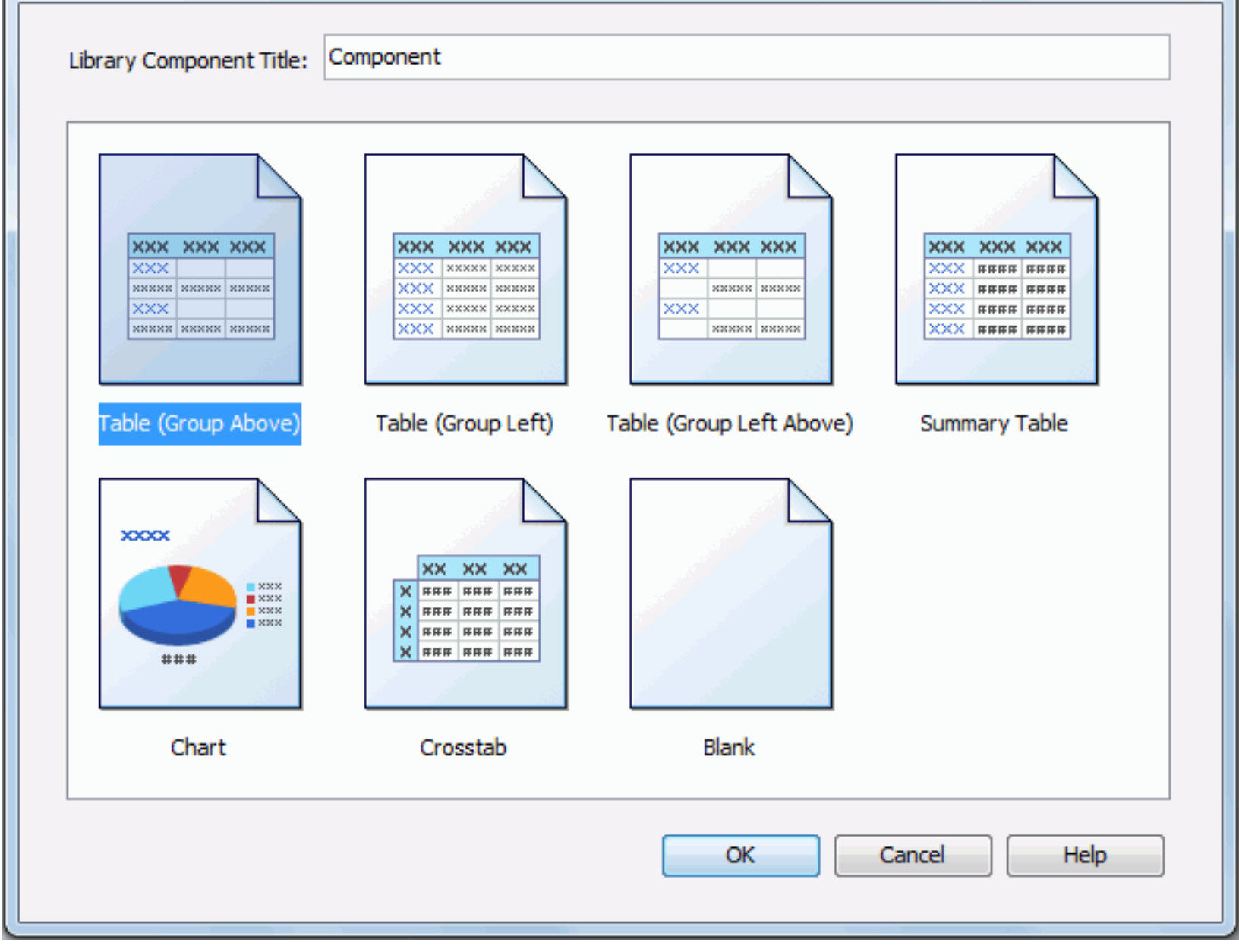
To create library components:

First the report developer should create library components to present data using JReport Designer and then publish them to JReport Server.

1. In JReport Designer, click **File > New > Library Component** on the menu bar to display the New Library Component dialog.



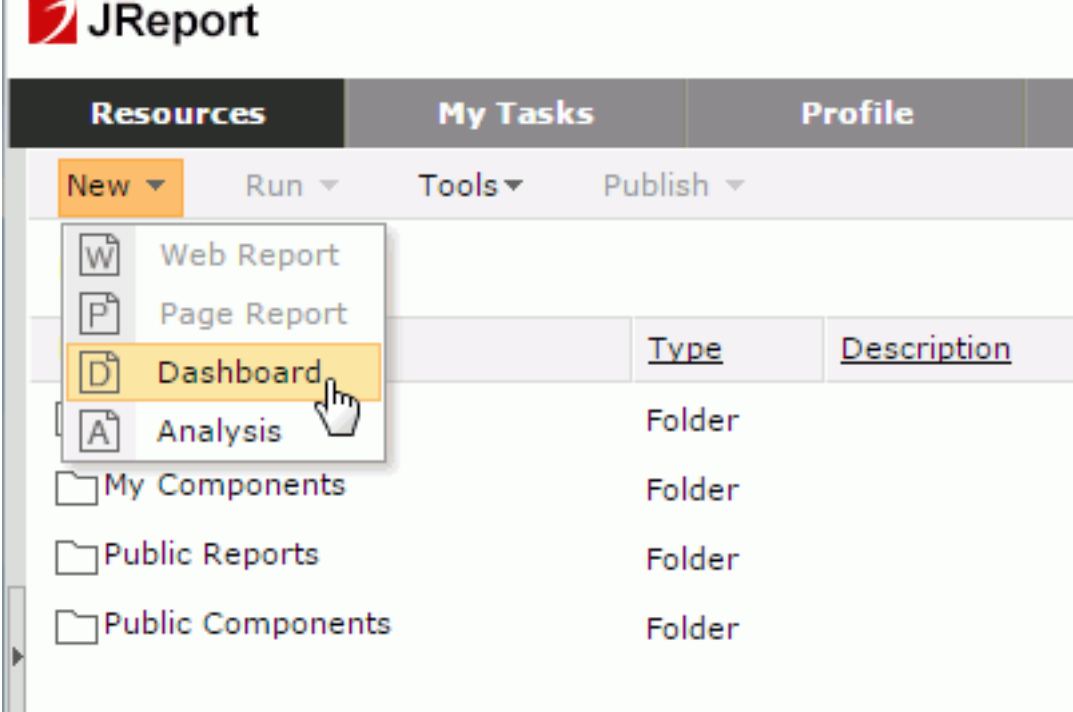
2. Input a title for the library component and select a layout with which to create the library component.



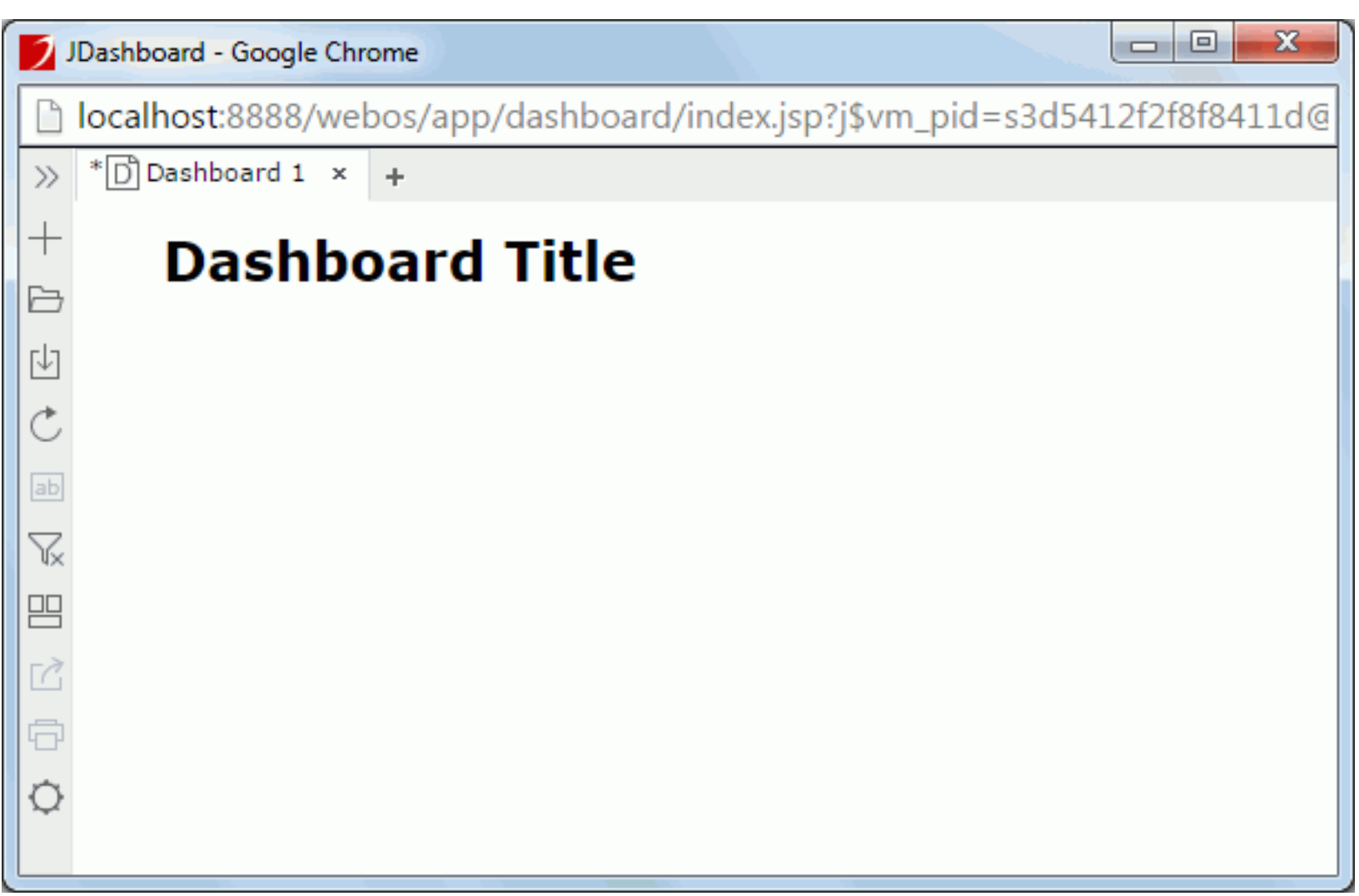
3. Click **OK**.
 - If Blank is selected as the layout, a report which is blank will be created. You can then use the Toolbox and the Resource View panels to add objects and view elements to the report.
 - If you select the layout as Table, Chart, or Crosstab, the corresponding wizard will then be displayed. Specify the settings according to your requirements.
4. If required, equip the library component with a configuration panel, which can be used to specify parameter values when there are parameters used in the library component, to filter or sort on the library component, or to change properties of the library component.
5. Create more library components following the above steps.
6. You can define message delivery between the library components, such as where and how to send a message, who will be able to receive the message and what to do after receiving.
7. Save the library components.
8. Publish the library component to JReport Server via the menu File > Publish and Download > Publish to Server > Publish Component to Server.

Then the server users will be able to access the published library components when creating dashboards using JDashBoard.

1. On the JReport Console > Resources page, click **New > Dashboard**. A blank dashboard will be created.

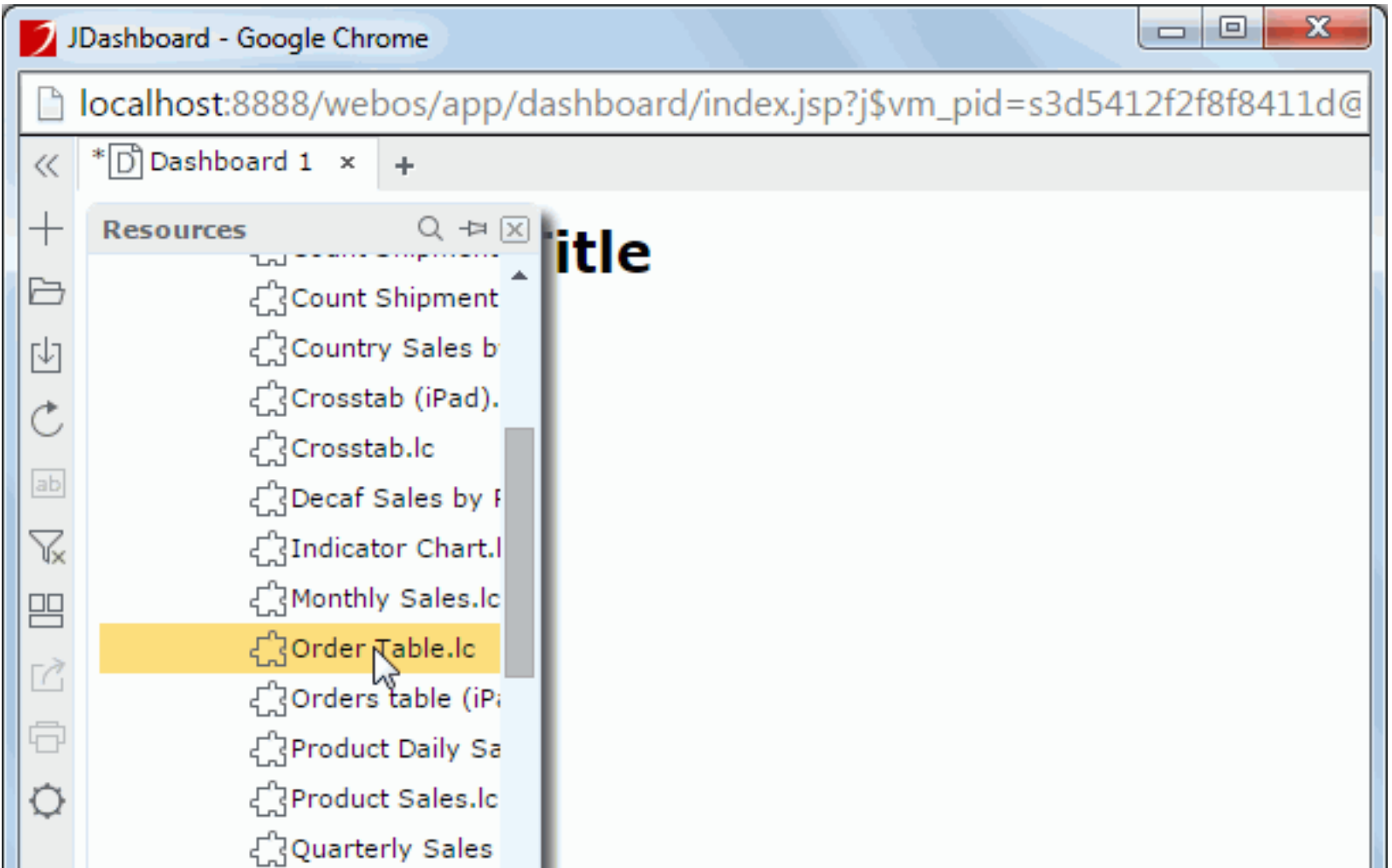


2. The dashboard header is the upper section of the blank editing area. You can make the bottom border of the header shown which is a dotted horizontal line by hovering the mouse over the header section. In the header you can insert labels, images, and special fields like Print Date and User Name via Toolbox in the Resources panel.

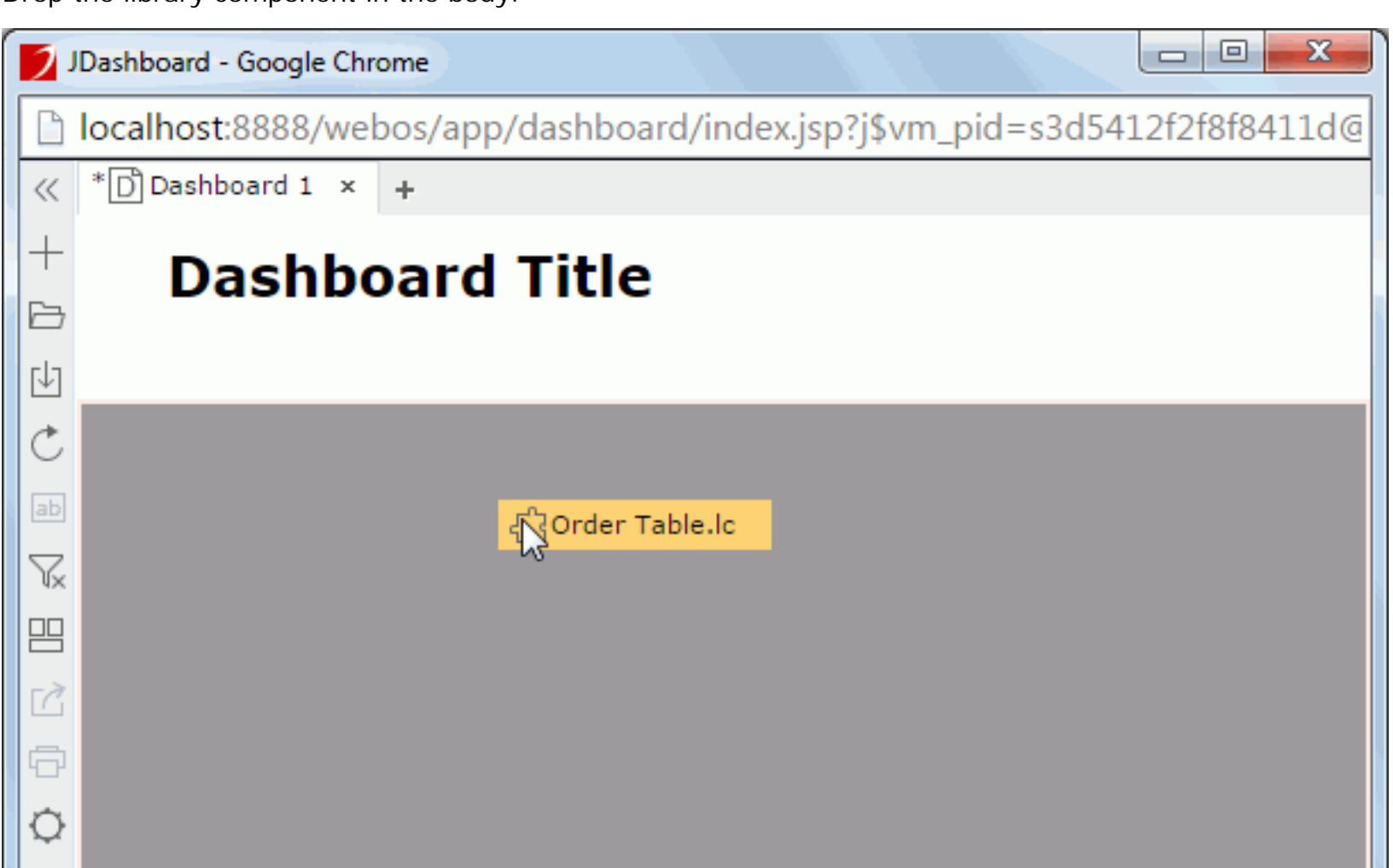


3. The dashboard body is the section below the header. In the body you can insert library components, sliders, and filter controls via the Resources panel.

Drag a library component from the Component Library:



Drop the library component in the body:



4. Save your dashboard.





Create Ad Hoc Reports

There are two types of ad hoc reports in JReport, Web Report and Page Report. Both are created from a business view in a catalog. Business views have the capability of providing multiple hierarchies allowing for automatic drill down and drill up for detailed analysis and slicing and dicing data which is especially powerful using a crosstab component.

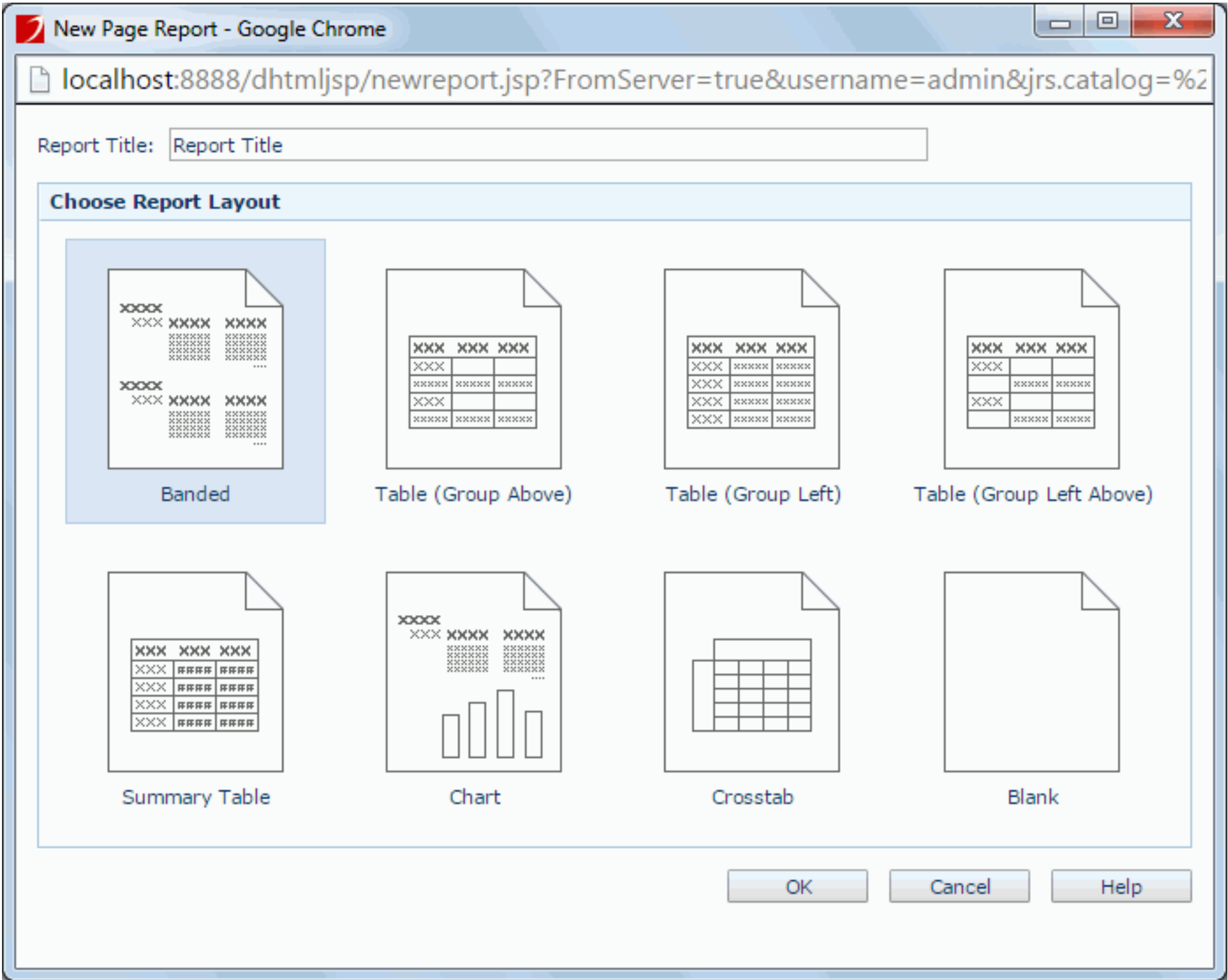
- Using Web Report Wizard the user can easily in a single pass create a multi-component report which is viewed in Web Report Studio. Web Report Studio provides the user a very high quality Web 2.0 Rich Internet Application (RIA) view of the report with component level navigation commands. For example, a single pass through the wizard can create a tabular style report with a chart and crosstab followed by the detail information in a table.
- Using Page Report Wizard the user can create a single component report which is viewed in Page Report Studio. Page Report Studio provides the user a page mode view of the report very similar to a printed report.

To create a Page Report:

- On the JReport Console > Resources page, open the folder into which the catalog that contains the predefined business views is published, then select the catalog from the Catalog drop-down list.
- Click **New** > **Page Report** on the task bar of the Resources page.



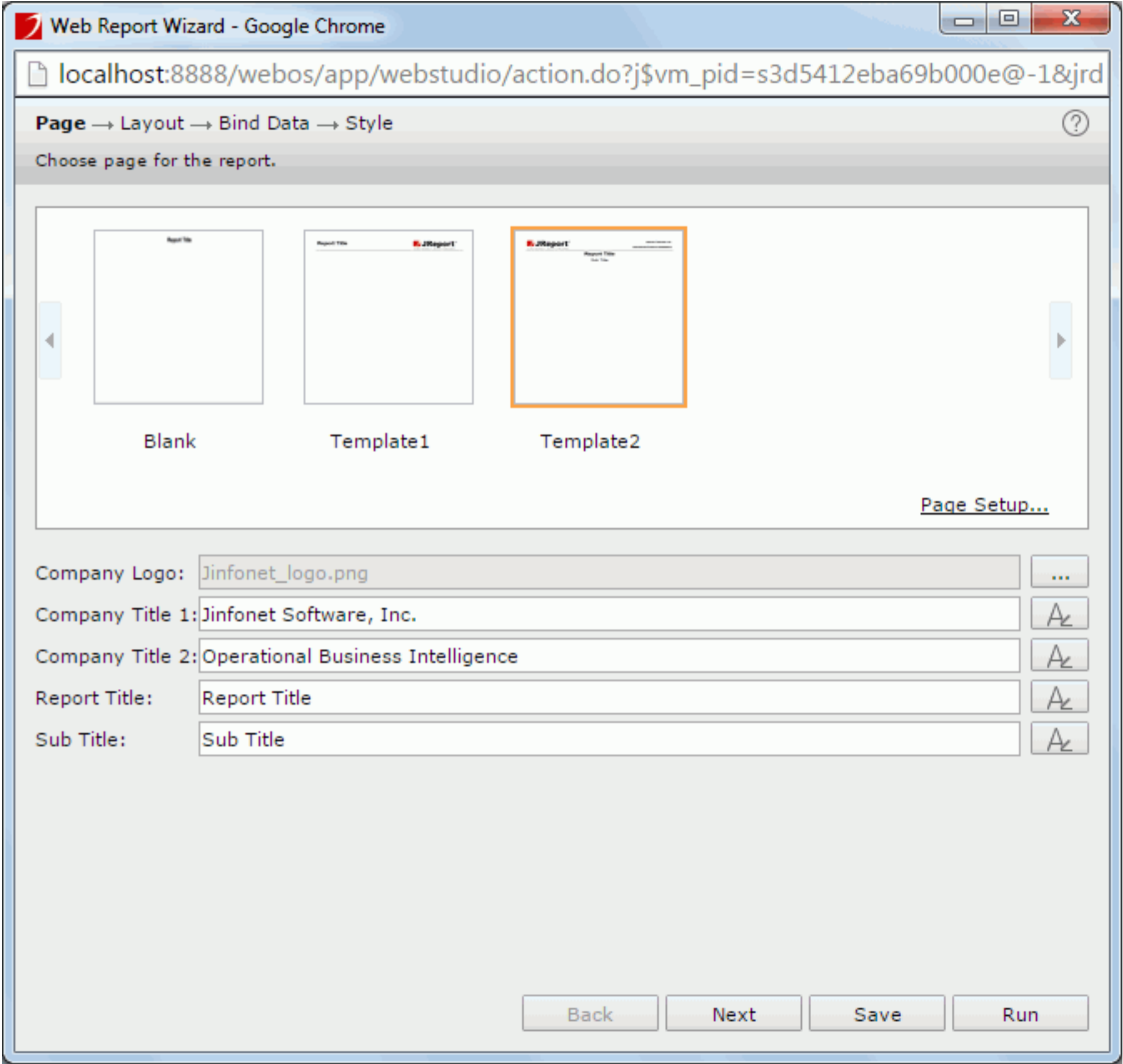
- The New Page Report dialog will appear for you to create a page report with the first report tab in it.



- Specify the title of the report tab as required in the Report Title text box.
- In the CHOOSE Report Layout box, select the required layout with which you want to create the report tab.
- Click **OK** to create the report.
 - If Blank Report is selected as the layout, a report 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.
 - 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.

To create a Web Report:

- On the JReport Console > Resources page, open the folder into which the catalog that contains the predefined business views is published, then select the catalog from the Catalog drop-down list.
- Click **New** > **Web Report** on the task bar of the Resources page.
- The Web Report Wizard is displayed. Follow the wizard to create the report.



Note: Before you can create a report in Page Report Studio, you need to first make sure that the Pop-up Blocker is not enabled on your web browser.



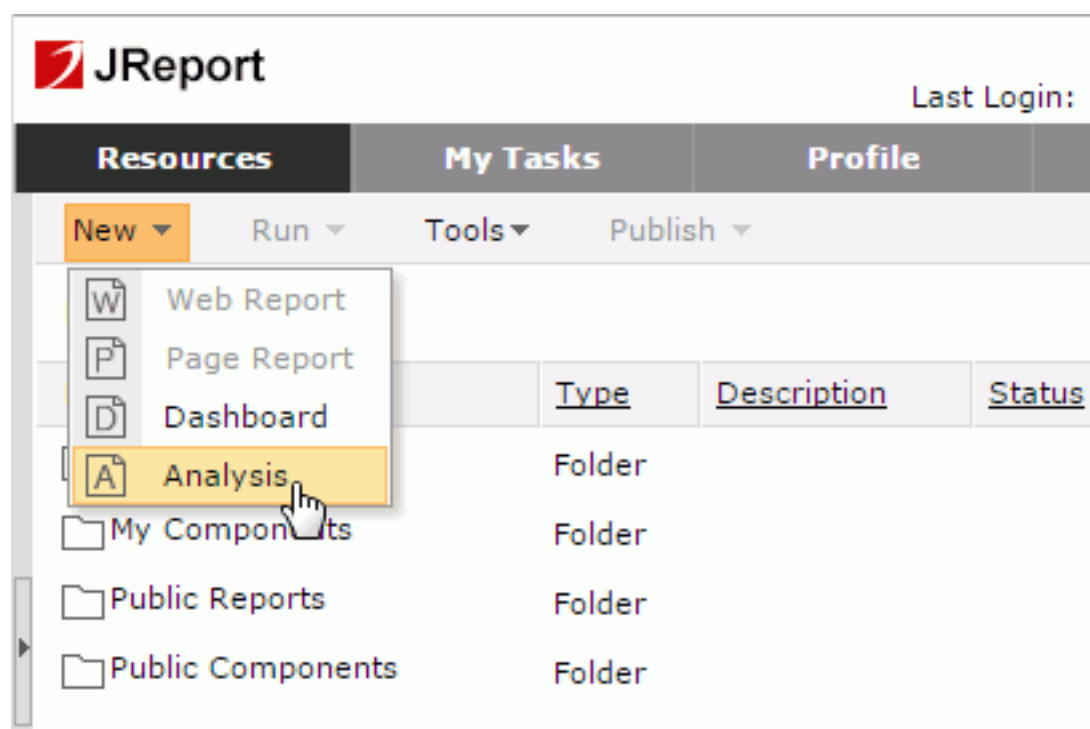


Start a Visual Analysis Session

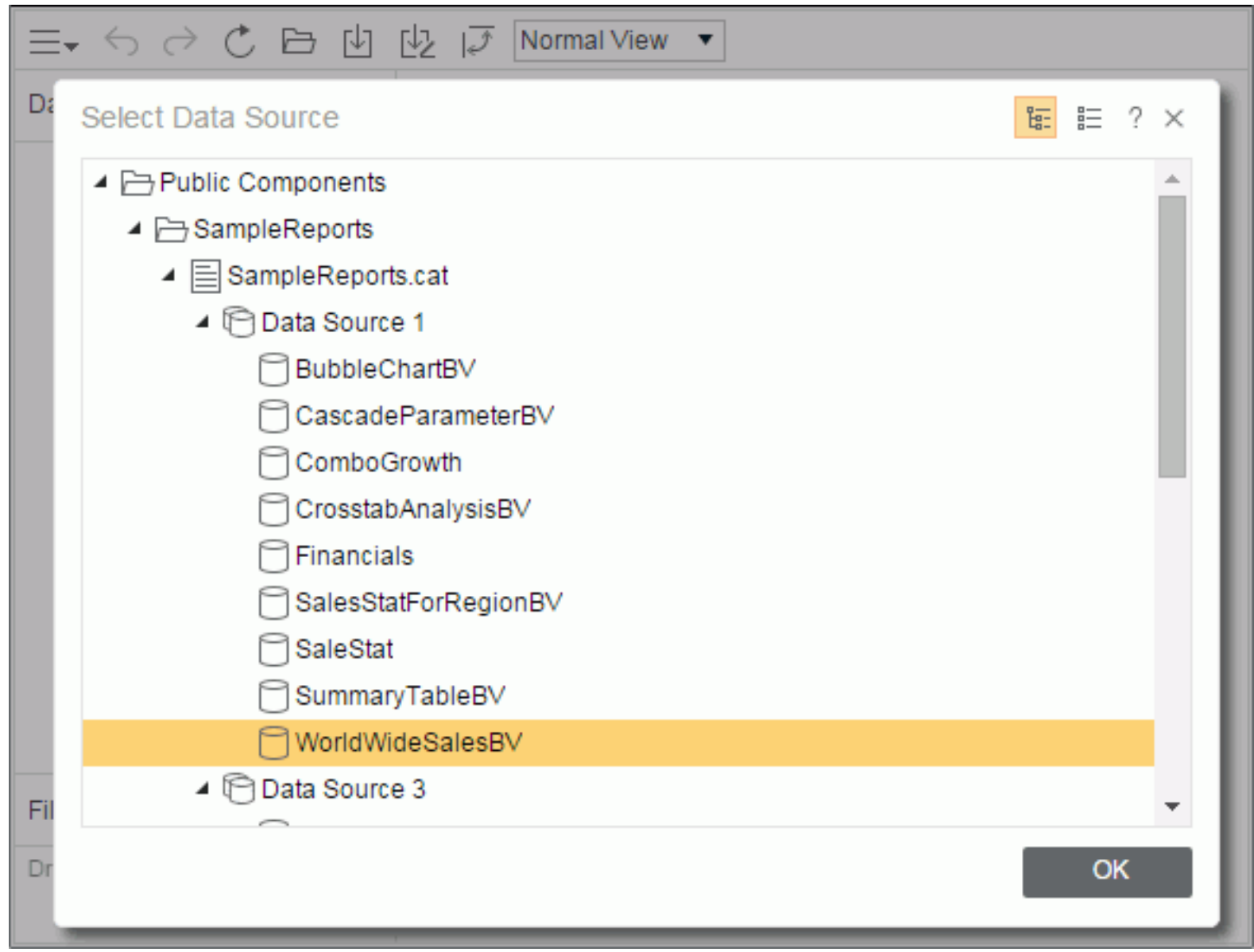
Visual Analysis is a WYSIWYG product to visualize the result of every step of your work. Simply by dragging and dropping data fields to the layout module, you are able to experience the detailed building up of crosstabs and charts step by step visually.


To start a new Visual Analysis session and perform data analysis:

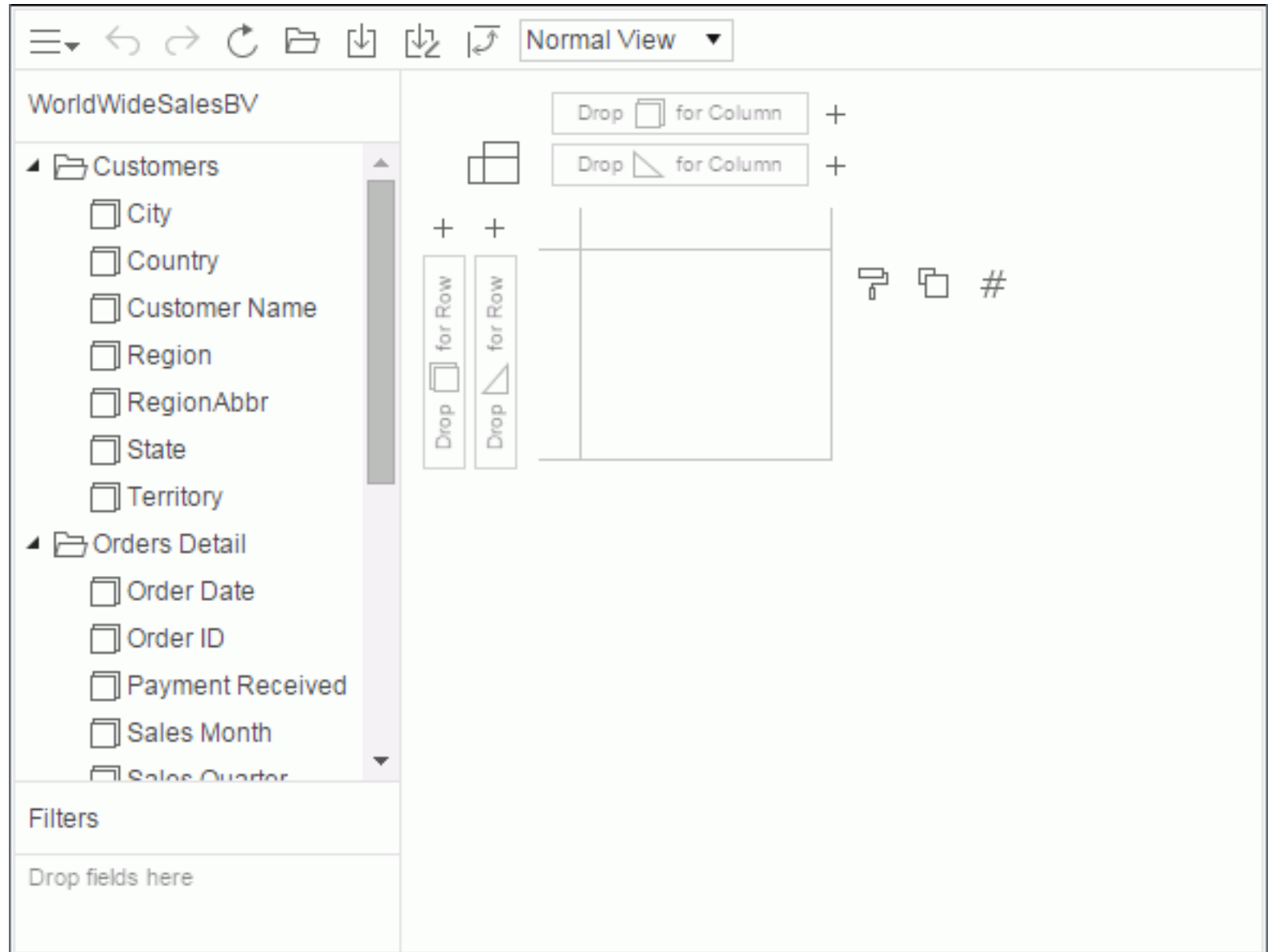
1. On the JReport Console > Resources page, click **New** > **Analysis**.



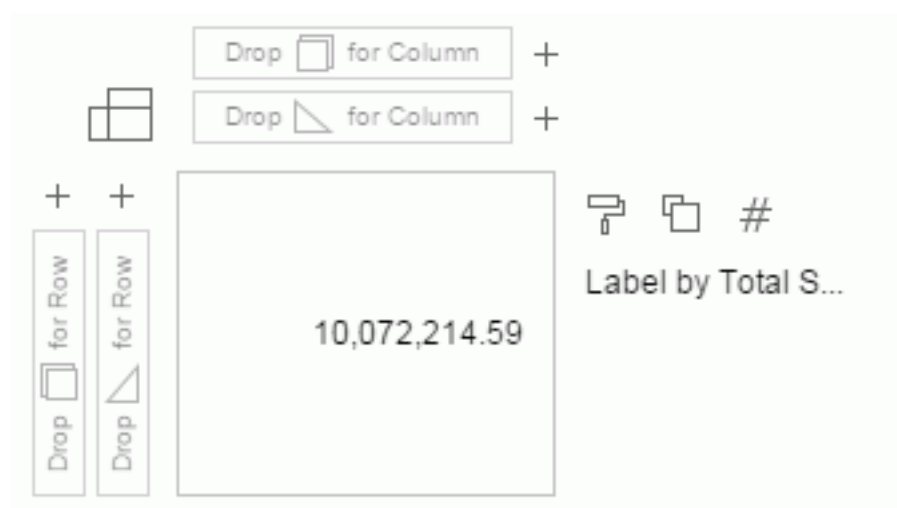
2. In the Select Data Source dialog, select the required one, for example, *WorldWideSalesBV* in / SampleReports, and then click **OK**.



3. The Visual Analysis window will be loaded. The display type shows Text  by default.

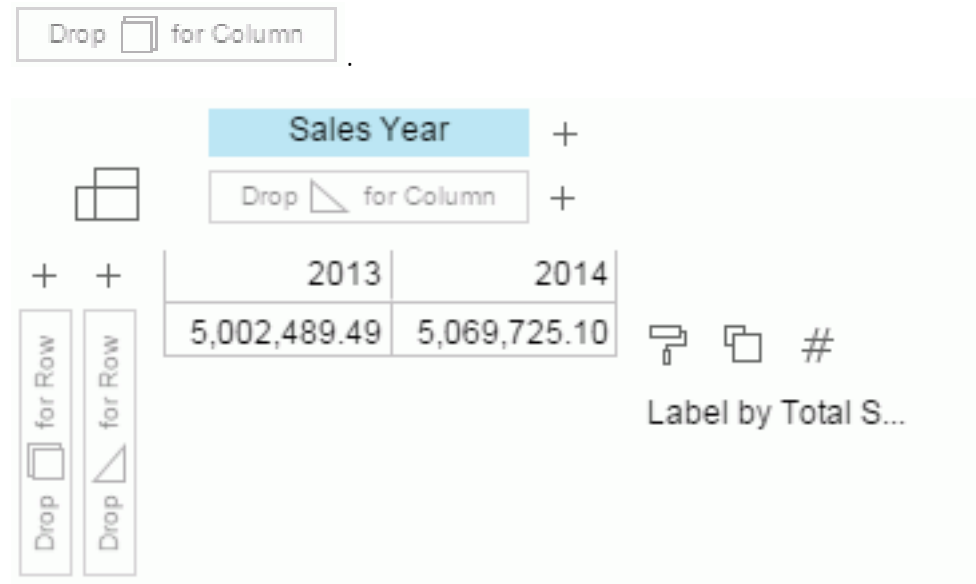


4. To view total sales, drag **Total Sales** from the Resources panel and drop it to the Label button  in the legend section.

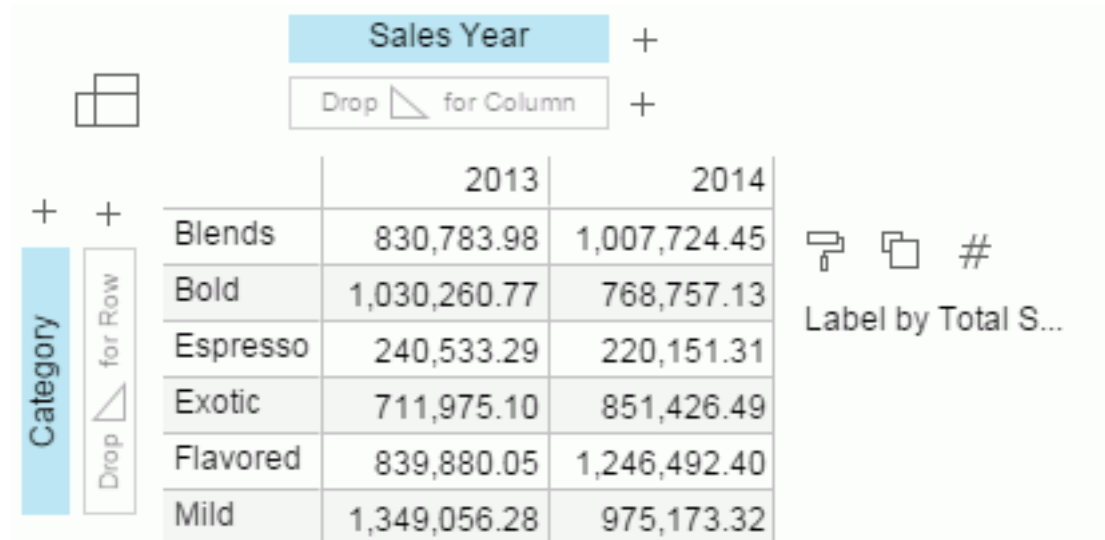



There is only one value in the data presentation area which is the total sales in the whole business view.

5. Now let's view total sales in different years. Drag **Sales Year** to the column control box



6. To add product category as the row header, drag **Category** to the row control box



7. Click  on the toolbar to save the result as a visual analysis template.



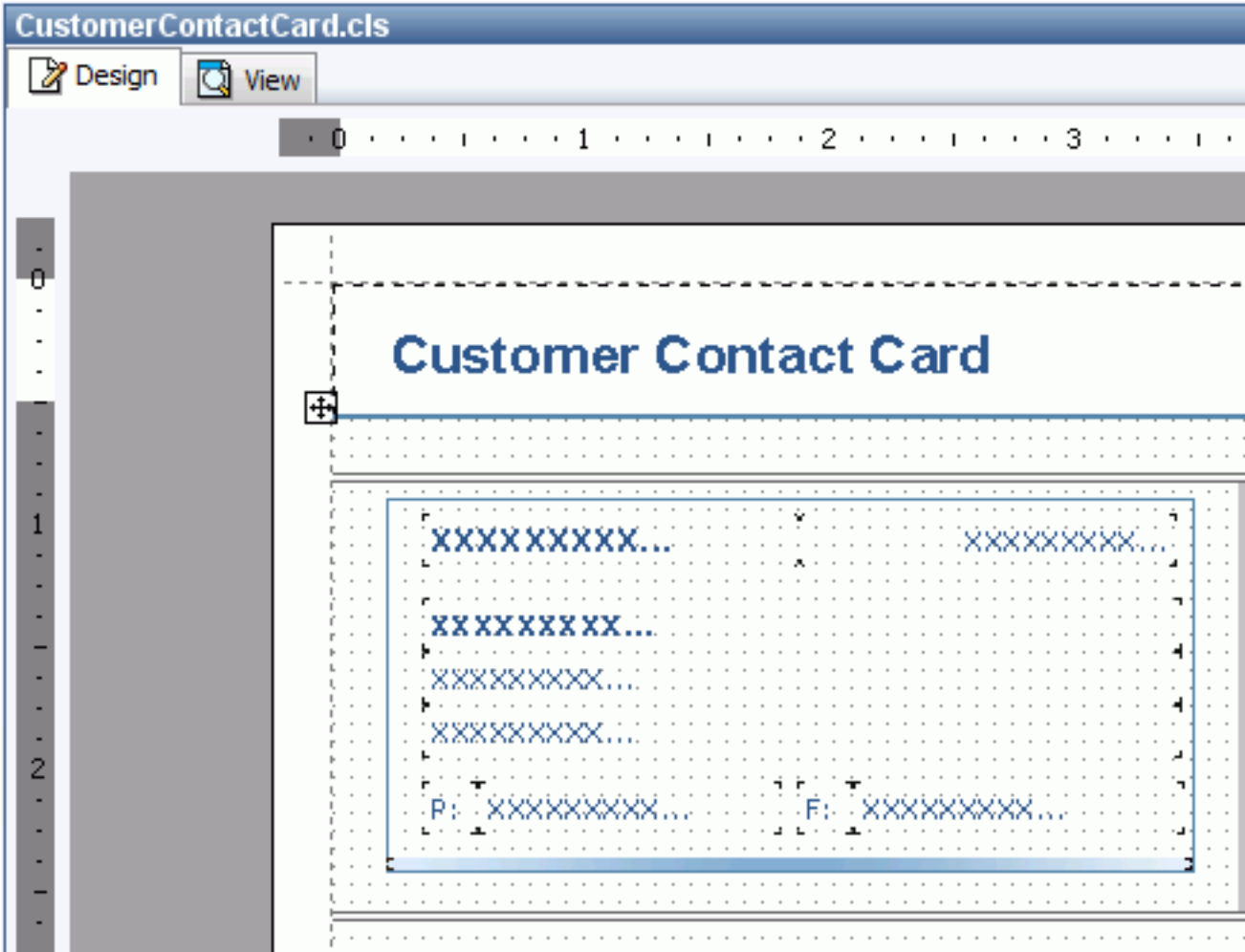


Open Sample Report Templates

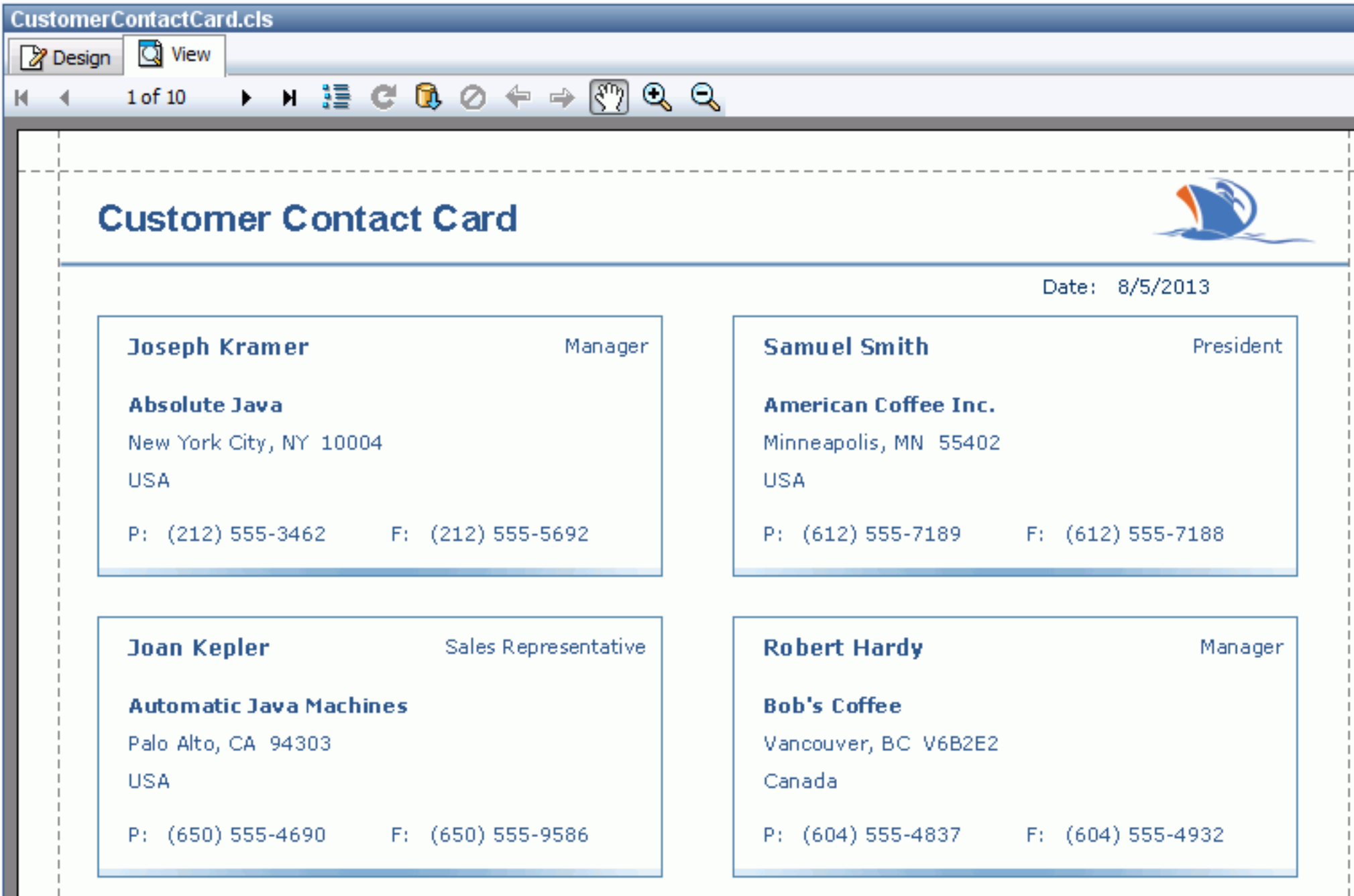
There are many sample reports in JReport Designer, you can open any of them to learn how to create them. There are two directories for sample reports and sample components. If you want to see how a particular component works such as a checkbox web control or a particular chart type such as scatter chart, open the catalog in <install_root>\Demo\Reports\SampleComponents. If you want examples of complete reports similar to what you will provide to your end users, open the catalog in <install_root>\Demo\Reports\SampleReports.

To open a sample report template, following the steps below:

1. Click **File** > **Open** in JReport Designer.
2. In the Open Report dialog, click the **Browse** button to open the catalog file SampleReports.cat in <install_root>\Demo\Reports\SampleReports.
3. Choose one of the reports listed, and then click **OK** to open it.



4. To preview the report result, click the **View** tab.





Start Creating New Reports

Enable data source

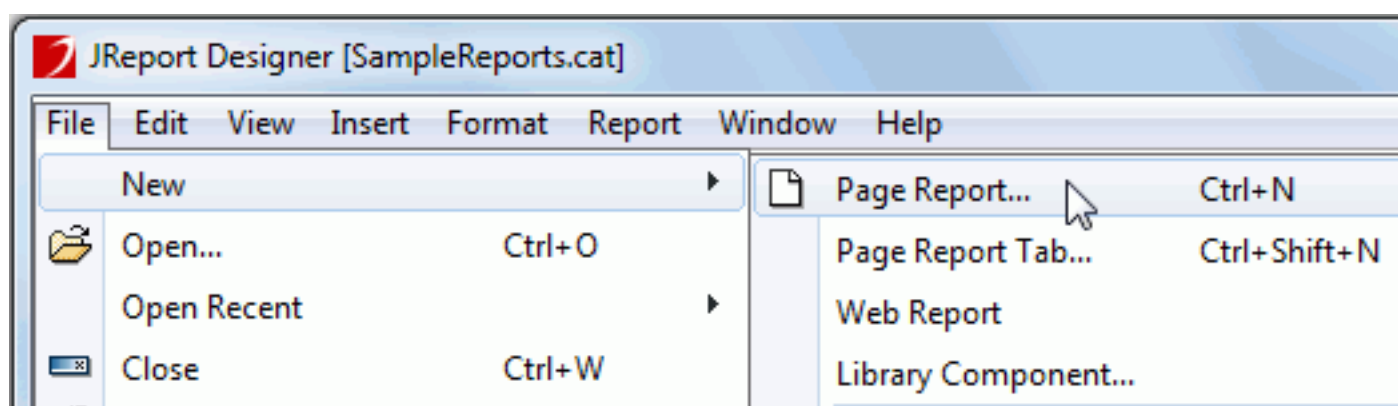
Before you create new reports using JReport Designer, you have to first create a catalog and then create a connection to your data source. Currently, seven methods are provided for you to achieve this goal: via JDBC/ODBC Connection, XML Connection, Web Service Connection, MongoDB Connection, HIVE Connection, Hierarchical Data Source, or by creating a User Defined Data Source.

- When the JDBC, XML, web service, MongoDB or HIVE connection is set up, you can then create queries based on tables in the connection, and then use the queries to create reports.
- The user defined data sources (UDS) and hierarchical data sources (HDS) can be used to create reports directly.
- Moreover, to change the property value of data objects, you should first click **File > Options** , and then uncheck **Forbid editing data object properties** in the Catalog category of the Options dialog.

Creating a page report

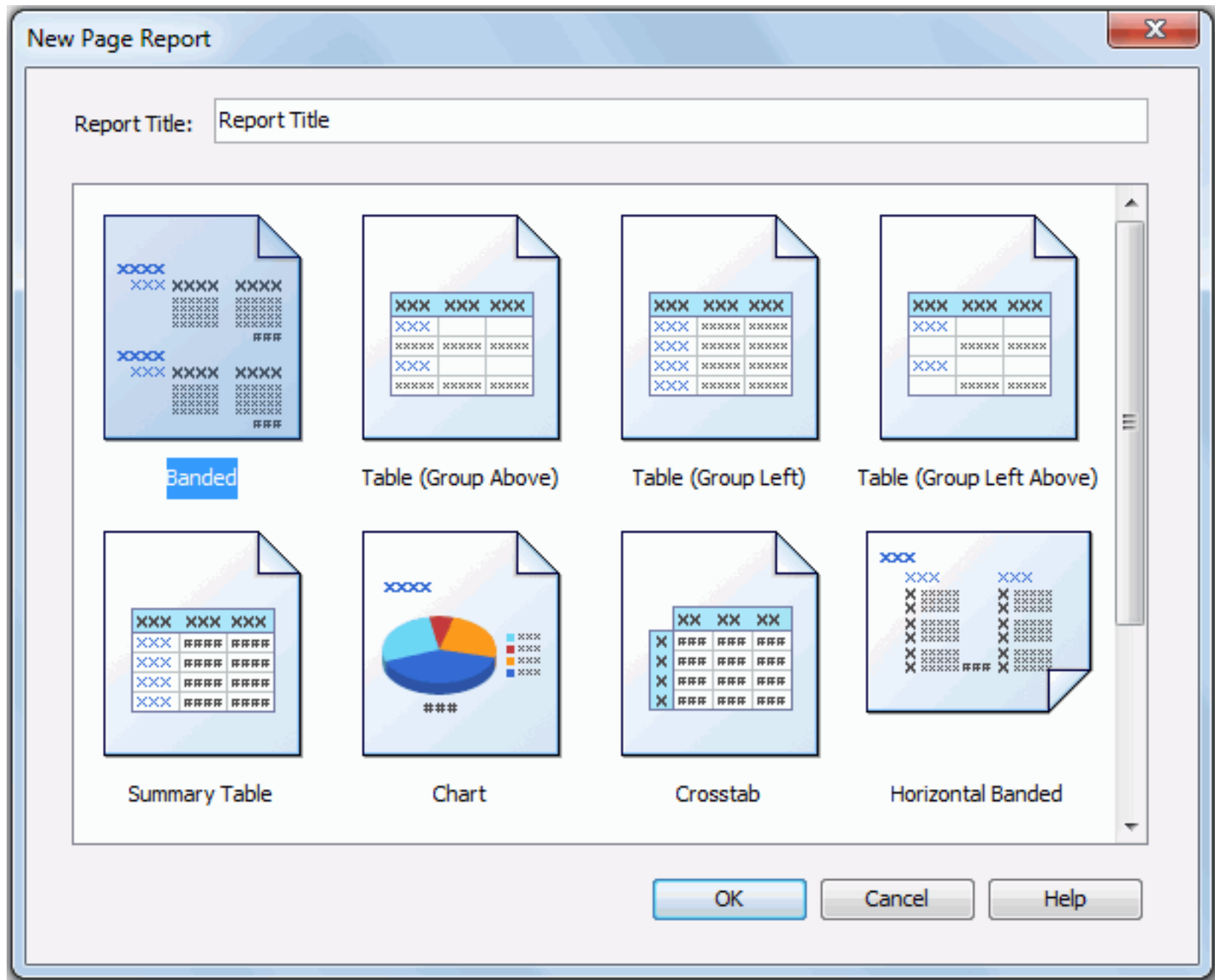
To create a page report, follow the steps below:

1. Start JReport Designer.
2. Click **File > New > Page Report** on the menu bar.



3. In the New Page Report dialog, select the layout of the first report tab in the report.

Since a page report cannot be empty, you will also need to create a report tab in it when creating the page report. See the section "Creating a page report tab" below for details about how to create the first report tab in a page report.



4. Click **OK** to create the report.

Creating a page report tab

Before you can create a page report tab, you first have to create a page report or open an existing page report.

JReport provides you with the following report layouts, which are designed for serving different reporting requirements:

- **Banded**
Creates a report containing a vertical 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.
- **Horizontal Banded**
Creates a report containing a horizontal banded object.
- **Mailing Label**
Creates a report containing a banded object in the form of a mailing label layout.
- **Tabular**
Creates a report containing a tabular component.
- **Blank**
Creates a report with nothing in it.

Creating a web report

To create a web report, follow the steps below:

1. On the JReport Designer menu bar, click **File > New > Web Report**.
2. A blank web report with a tabular of one cell is displayed. You can then split the tabular, insert components to the tabular cells, and modify the component properties as you want.



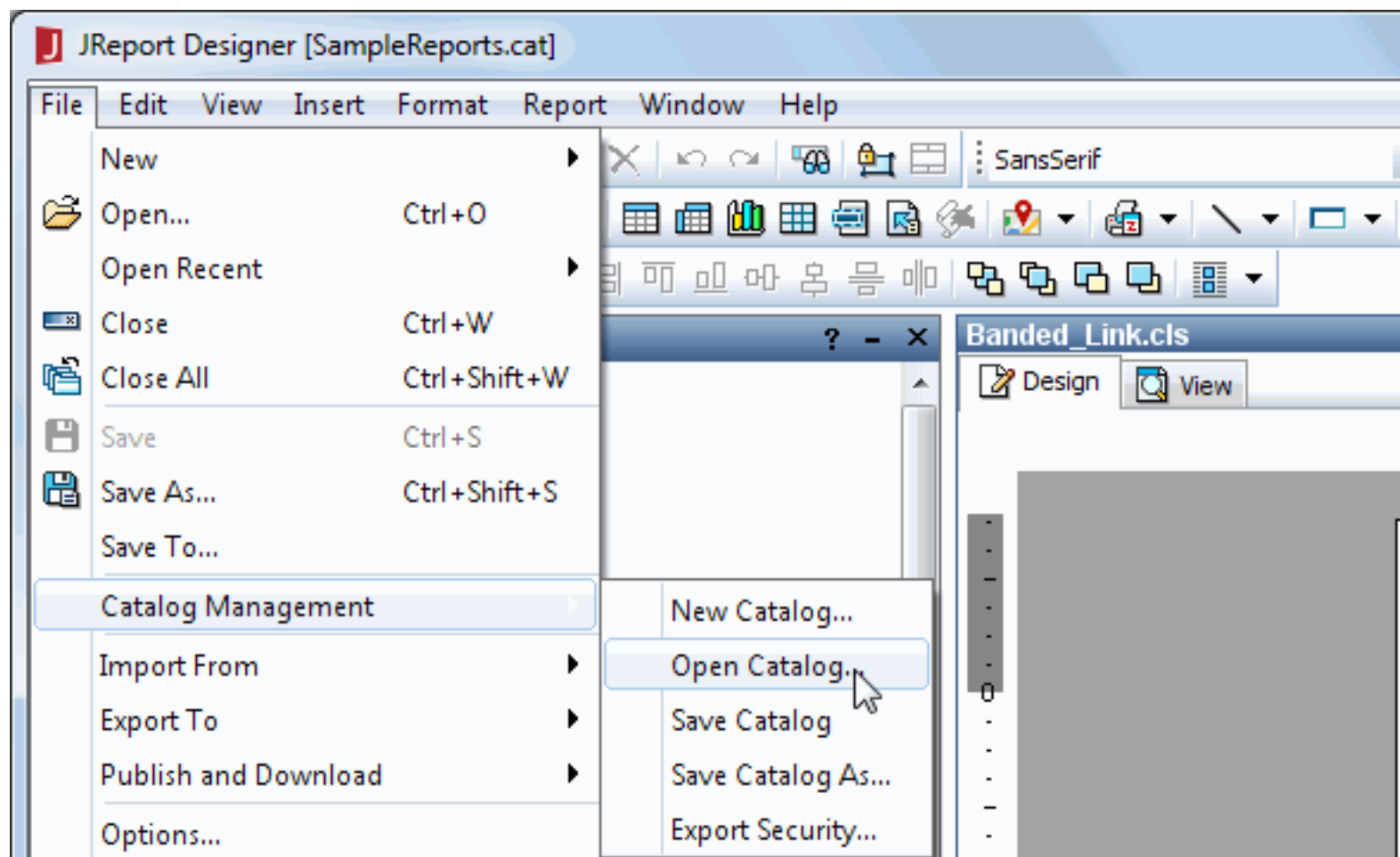
Build Business Views

Business views provide report users with a business-oriented view, and calculations on a data source. They shield report end users from having to understand physical structure, and enables the easy building of reports.

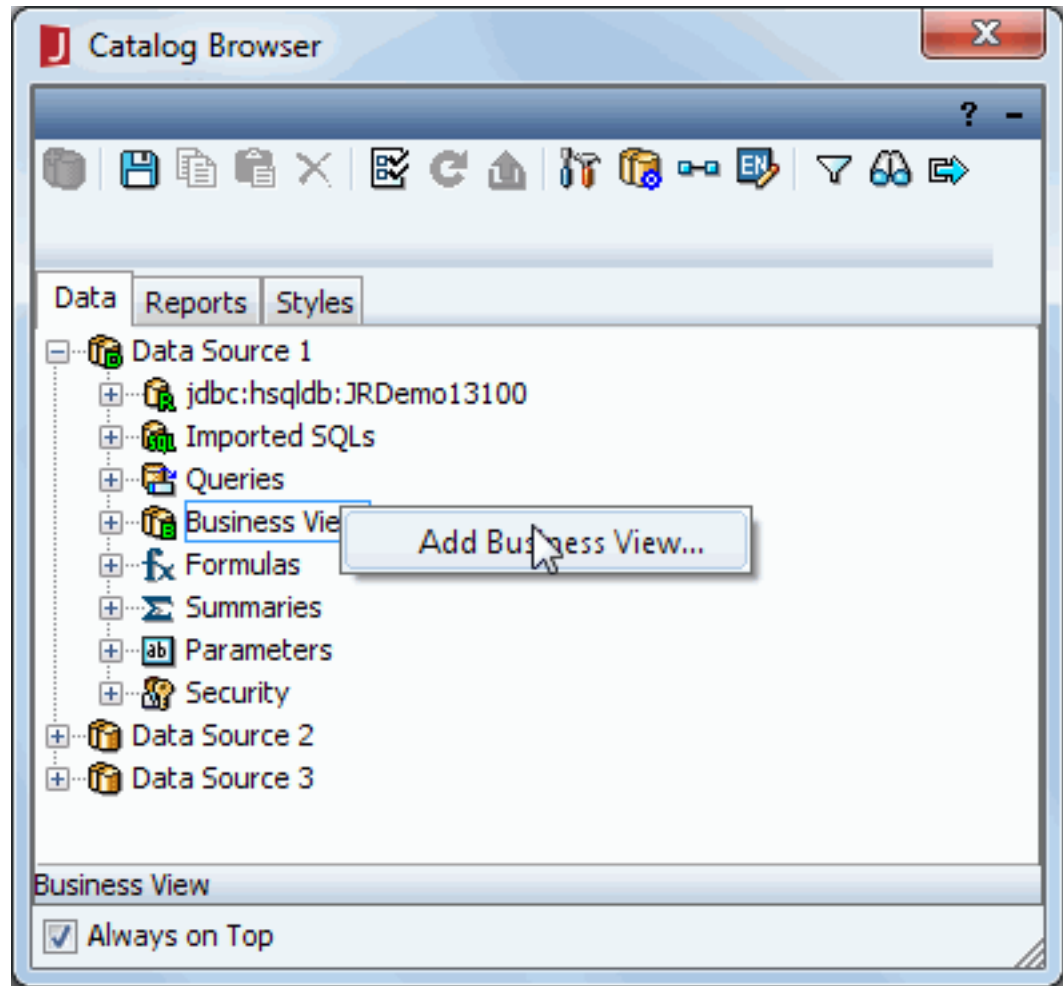
To make use of business views in Page Report Studio and Web Report Studio, you need to first define them at report design time in JReport Designer.


To add a business view to a catalog:

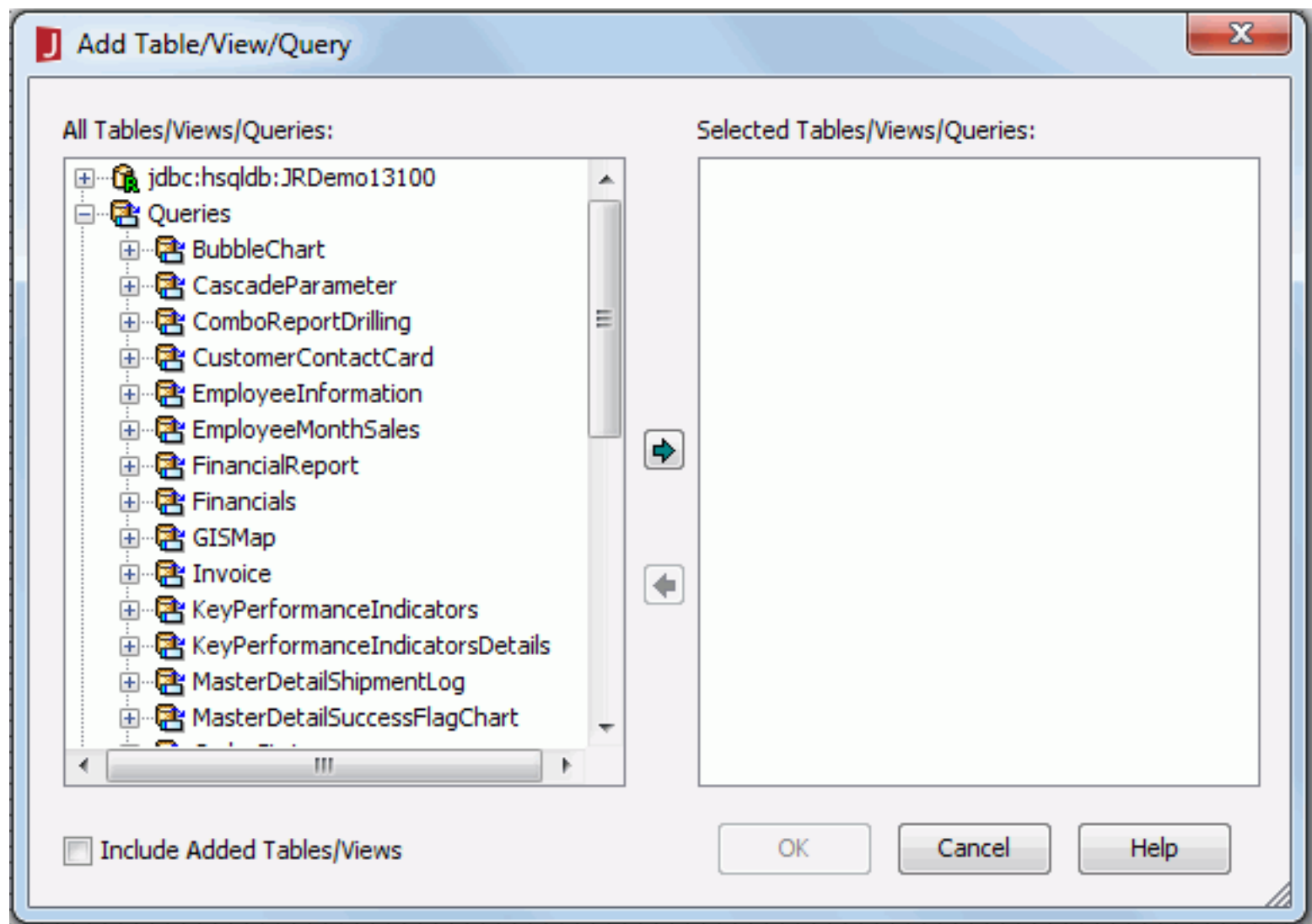
1. In JReport Designer, click **File > Catalog Management > Open Catalog** to open the catalog to which you want to add the business view.



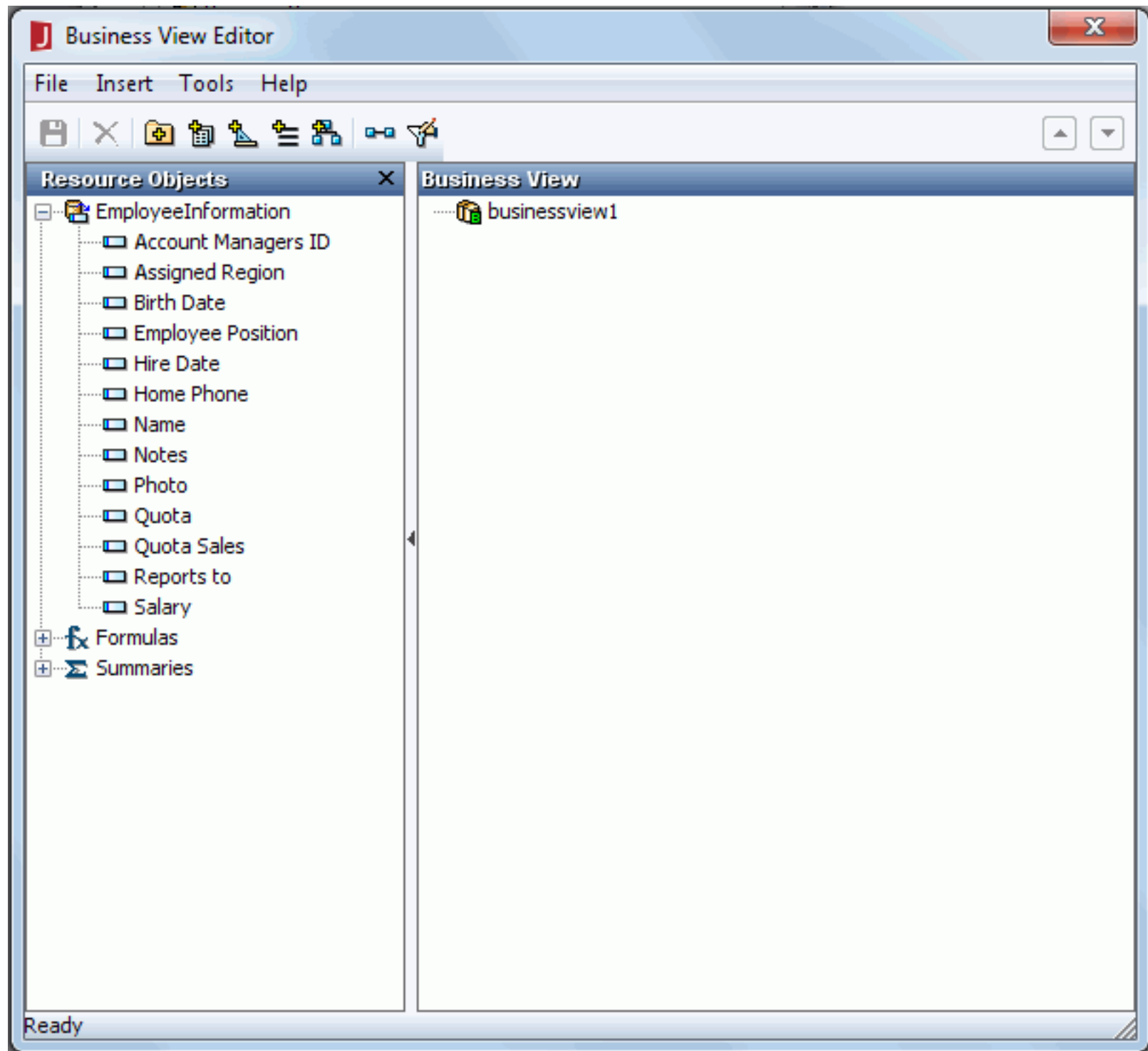
2. In the Data tab of the Catalog Browser, expand the desired data source, right-click the **Business View** node and then click **Add Business View**.



3. In the Input Business View Name dialog, enter a name for the business view, and then click **OK**.
4. In the Add Table/View/Query dialog, select a resource you want from the left box and click  to add them into the right box. Click **OK**.



5. In the Business View Editor window, add elements to the business view.



6. Define hierarchies on the business view to allow end users to drill report data to particular groups.
7. Create some predefined filters for the business view to narrow down records if required.
8. Configure security for the business view if necessary.
9. Click **File > Save** to save the business view.
10. Click **File > Close** to close the window. The business view will now have been added to the catalog.





Online Help

Online documentation includes all information about the JReport products. You can go <http://www.jinfonet.com/jreport-documentation> to learn about it.

