



# **JASPERREPORTS SERVER COMMUNITY PROJECT USER GUIDE**

RELEASE 4.1

<http://www.jaspersoft.com>

---

Copyright © 2011 JasperSoft Corporation. All rights reserved. Printed in the U.S.A. JasperSoft, the JasperSoft logo, JasperAnalysis, JasperServer, JasperETL, JasperReports, JasperReports Server, and iReport, are trademarks and/or registered trademarks of JasperSoft Corporation in the United States and in jurisdictions throughout the world. All other company and product names are or may be trade names or trademarks of their respective owners.

This is version 0611-JS041-3 of the *JasperReports Server Community Project User Guide*.

## TABLE OF CONTENTS

---

<b>Chapter 1</b>	<b>Introduction to JasperReports Server</b>	<b>7</b>
1.1	Logging In	8
1.2	Browsing the Repository	8
1.3	Searching the Repository	10
1.3.1	Searching the Entire Repository	10
1.3.2	Filtering Search Results	10
1.4	Using Repository Resources	12
1.5	Moving Folders	13
1.6	Sorting the Repository List	13
1.7	Mastering Report Design	14
1.7.1	Locating JasperReports Samples	14
1.7.2	Learning about the Samples	14
<b>Chapter 2</b>	<b>Running Reports</b>	<b>15</b>
2.1	Running a Simple Report	15
2.2	Running a Report with Input Controls or Filters	16
2.2.1	Simple Input Controls	17
2.2.2	Cascading Input Controls	18
2.3	Scheduling Reports	19
2.3.1	Setting Up a Job	20
2.3.2	Setting Output Options	21
2.3.3	Viewing the List of Scheduled Jobs	23
2.3.4	Changing Job Schedules	23
2.3.5	Stopping a Job from Running	24
2.3.6	Running a Job Repeatedly	24
2.3.7	Running a Job in the Background	26
2.3.8	Adding a Date/Time Stamp to Scheduled Output	26
2.4	Event Messages	27

<b>Chapter 3</b>	<b>Adding Reports Directly to the Repository</b>	<b>29</b>
3.1	Overview of a Report Unit	29
3.2	Adding a Simple Report Unit to the Server	30
3.2.1	Uploading the Main JRXML	30
3.2.2	Uploading Suggested File Resources	32
3.2.3	Defining the Data Source	34
3.2.4	Defining the Query	35
3.2.5	Saving the New Report Unit	37
3.3	Adding a Complex Report Unit to the Server	38
3.3.1	Uploading Undetected File Resources	40
3.3.2	Adding Input Controls	42
3.3.3	Selecting a Data Source and Running the Complex Report	52
3.4	Adding Cascading Input Controls to a Report	54
3.5	Editing JRXML Report Units	55
3.6	Localizing Reports	56
3.6.1	Running a Localized Report	56
3.6.2	Adding a Multi-Lingual Prompt and Static List to an Input Control	57
3.6.3	Reusing Resource Bundle	66
3.6.4	Using Default Fonts in JasperReports Server	66
<b>Chapter 4</b>	<b>Administration</b>	<b>67</b>
4.1	Managing JasperReports Server	67
4.2	Managing Users	67
4.3	Managing Roles	71
4.4	Access Control	75
4.4.1	Authentication Overview	75
4.4.2	Authorization Overview	75
4.4.3	Permissions	76
4.4.4	Assigning Permissions	78
4.4.5	Testing User Permissions	80
4.5	Configuring Password Options	80
4.5.1	Enabling Password Expiration	80
4.5.2	Allowing Users to Change their Passwords	81
<b>Chapter 5</b>	<b>Themes</b>	<b>83</b>
5.1	Introduction	83
5.2	How Themes Work	84
5.2.1	Theme Files	85
5.2.2	Inheritance Mechanism	85
5.2.3	CSS Priority Scheme and Custom Overrides	86
5.3	Administering Themes	86
5.3.1	Setting the Active Theme	86
5.3.2	Creating Theme Folders and Files	88
5.4	Working With CSS Files	90
5.4.1	Theme Development Workflow	90

5.4.2	Firebug Plug-in for FireFox .....	90
5.4.3	Test Platform .....	91
5.4.4	User Interface Samples .....	91
<b>Chapter 6</b>	<b>Accessing Reports from iReport .....</b>	<b>93</b>
6.1	Plug-in Requirements .....	93
6.2	Connecting to the Server from iReport .....	94
6.3	Creating a Report Unit in the Repository from iReport .....	95
6.3.1	Adding an Image and Subreport to the Report Unit .....	97
6.4	Managing Resources in the Repository from iReport .....	99
6.5	Editing a Report in the Repository from iReport .....	100
6.6	Running a Report from iReport .....	101
<b>Glossary</b>	<b>.....</b>	<b>103</b>
<b>Index</b>	<b>.....</b>	<b>111</b>



## CHAPTER 1 INTRODUCTION TO JASPERREPORTS SERVER

---

The JasperReports Server Community Project builds on JasperReports as a comprehensive family of Business Intelligence (BI) products, providing robust static and interactive reporting, report server, and data analysis capabilities. These capabilities utilize common metadata and provide shared services, such as security, a repository, and scheduling. The server exposes comprehensive public interfaces enabling seamless integration with other applications and the capability to easily add custom functionality.

In a nutshell, JasperReports Server provides the ability to:

- Efficiently and securely manage many reports.
- Interact with reports, including entering parameters and drilling on data.
- Schedule reports for distribution via email and storage in the repository.

For business intelligence users, we offer Jaspersoft OLAP, which runs on the server. This component is described in its own user guide.

Jaspersoft provides several other sources of information to help extend your knowledge of JasperReports Server:

- Our Ultimate Guides document advanced features and configuration. They also include best practice recommendations and numerous examples. The guides are available as downloadable PDFs. Community project users can purchase individual guides or bundled documentation packs from the Jaspersoft [online store](#). Commercial customers can download them freely from the [support portal](#).
- Our free samples, which are installed with JasperReports, iReport, and JasperReports Server, are documented online. For more information about the samples, see section 1.7, “**Mastering Report Design**,” on page 14.
- JasperForge, our [community website](#), hosts open source projects, associated source code, tools for bug tracking, version control, and forums for community discussion. You'll find development and implementation advice, a secure development environment for community-driven projects, and community feedback.

This chapter contains the following sections:

- **Logging In**
- **Browsing the Repository**
- **Searching the Repository**
- **Using Repository Resources**
- **Moving Folders**
- **Sorting the Repository List**
- **Mastering Report Design**

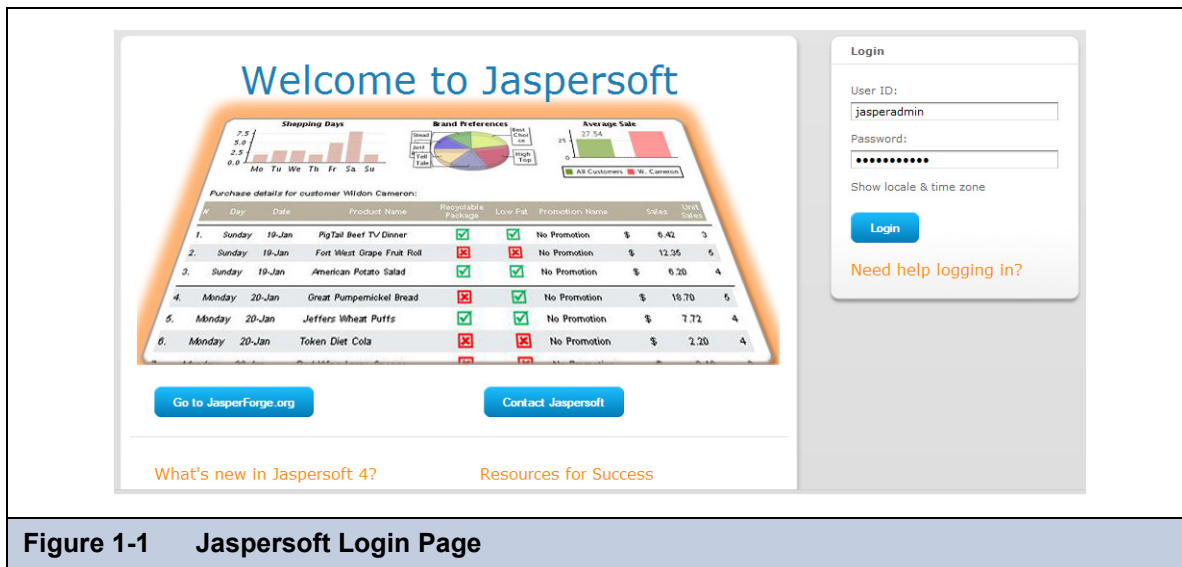
## 1.1 Logging In

To protect the data that you can access through the server, all users are required to log in with a password. This section includes a tutorial about logging into the server. The tutorials in this guide assume you've installed the sample data provided with the server.

### To log into the server:

1. Enter `http://hostname:8080/jasperserver` in your browser (where `hostname` is the name of the computer that hosts JasperReports Server).

The Jaspersoft Login page appears.



**Figure 1-1 Jaspersoft Login Page**

2. To log in, enter your user ID and password or a sample user name and password.



If you installed the sample data, you can log in using sample user names and passwords. For more information, click **Need help logging in?** The default administrator user name and password is `jasperadmin`; you should change it after installation.

3. If you want to use a different locale and time zone than the server uses, click **Show locale & timezone**.  
The **Locale** and **Time Zone** controls appear in the Login panel.
4. Select your locale and time zone from the drop-down menus.
5. Click **Login**.

If you entered a valid user name and password, the server displays the repository page.

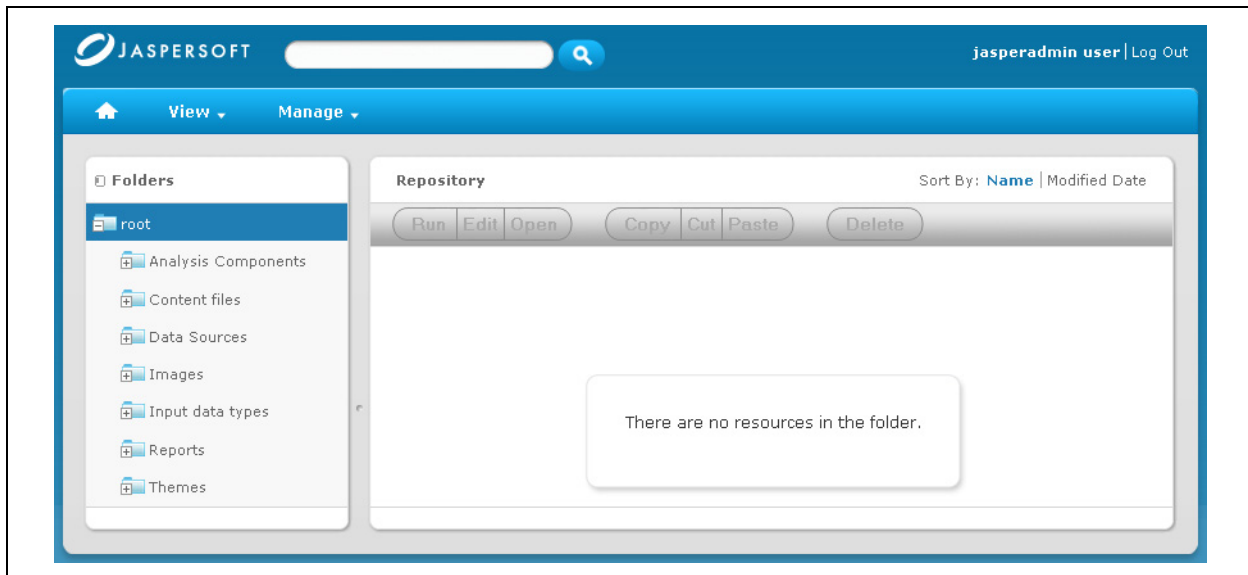
## 1.2 Browsing the Repository

The repository is the server's internal storage for reports, analysis views, and related files. The repository is organized as a structure of folders containing resources, much like a file system. However, unlike a file system, the repository is stored as a private database that only JasperReports Server can access directly.

To browse the repository, select **View > Repository**. From the repository page, you access the reports, themes, and other files stored on the server.





**Figure 1-2** shows the Repository page. The root directory, visible to administrative users, is empty. Expand and collapse folders in the **Folders** panel to find resources.



**Figure 1-2 Repository Folders Panel**

You can search the repository by entering text in the field at the top of the page. After finding a resource you can perform operations on it, subject to your access permissions. Typically, an administrator or resource owner sets permissions to access folders and other repository resources.

Logged in as a user, you only see the View menu. Logged in as an administrator, you see the View menu and the Manage menu. The  and the View and Manage menus offer these options:

Menu	Description
	Returns to the Repository page.
<b>View</b>	<ul style="list-style-type: none"> <li>♦ <b>Search Results</b> – Displays the repository of resources filtered by criteria selected in the Filters panel.</li> <li>♦ <b>Repository</b> – Displays the repository of files and folders containing resources, such as reports, report output, data sources, and images.</li> <li>♦ <b>Reports</b> – Lists the repository of reports.</li> <li>♦ <b>OLAP Views</b> – Accesses the repository of analysis views that you can view, run, and edit, depending on permissions. Only available with Jaspersoft OLAP.</li> <li>♦ <b>Messages</b> – Lists system messages, such as an error in a scheduled report.</li> <li>♦ <b>Samples</b> – Presents galleries of UI components that you redesign using Themes. Only available to administrators.</li> </ul>
<b>Manage</b>	<p>Only available if you're logged in as an administrator.</p> <ul style="list-style-type: none"> <li>♦ <b>Users</b> – Adds, deletes, enables, and edits user accounts and displays user properties.</li> <li>♦ <b>Roles</b> – Adds, deletes, and edits roles and displays role properties.</li> <li>♦ <b>OLAP Settings</b> – Sets OLAP Engine properties. Only available with Jaspersoft OLAP. For more information, see the <i>Jaspersoft OLAP User Guide</i>.</li> <li>♦ <b>Log Settings</b> – Sets the logging level, which determines how much information the server logs. Only available to administrators.</li> </ul>

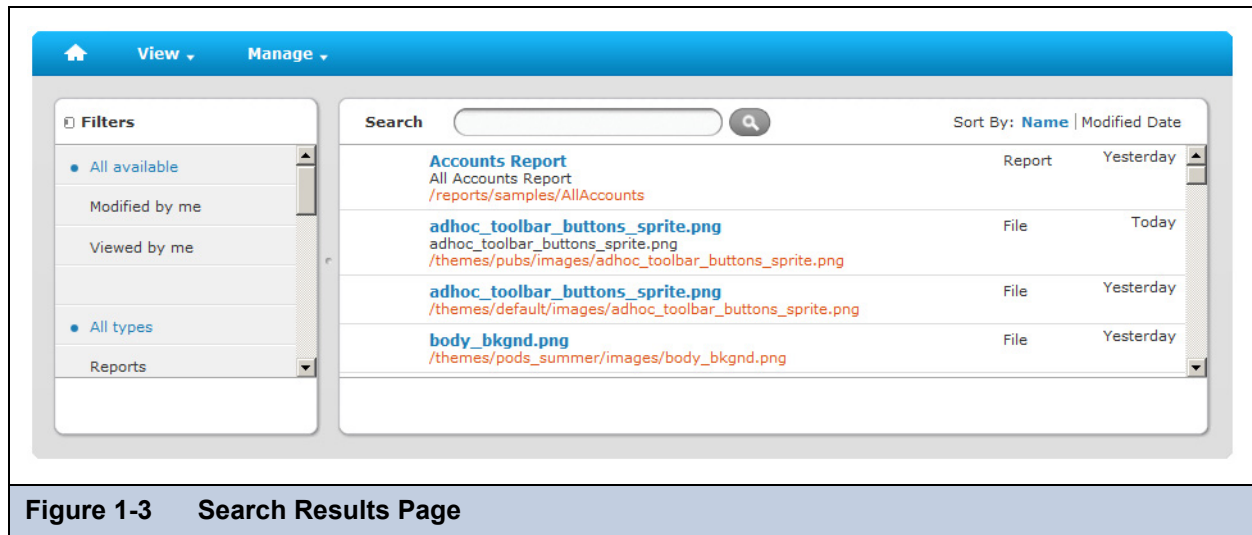
## 1.3 Searching the Repository

You can search the entire repository, subject to permissions, or narrow the search using filters. **Filters** restrict a search by name, who changed the resource, type of resource, date of the resource, and schedule.

### 1.3.1 Searching the Entire Repository



To search the repository, select **View > Search Results**. The search results page appears. Instead of only viewing resources by folder, use intuitive search criteria, such as who modified the resource and when, to find pinpoint resources.

On the search results page, use either the **Filters** panel or **Search** field to find resources. The search results page displays results of searches and filters.



**Figure 1-3 Search Results Page**


To search all resources in the repository:

1. Select one of these filters: **All available**, **Modified by me**, or **Viewed by me**.
2. Click the  icon in the search field to clear the search term if there is one.
3. Select **All types**, as shown in [Figure 1-3](#).
4. Click .

The search results appear, listing files that your user account has permission to view. Click a resource in the list to view it or right-click a resource to see what functions are available from the context menu.

The server remembers your settings on the Search Results page, making the most commonly needed resources remain visible when you return to the page.

### 1.3.2 Filtering Search Results

If you enter a search term and click  at the top of any server page, the server doesn't use filters. The search uses these default settings:

- Include subfolders
- Start at the top-most folder visible to the user
- Search for reports, report outputs, OLAP views, or other resources
- Sort alphabetically by name


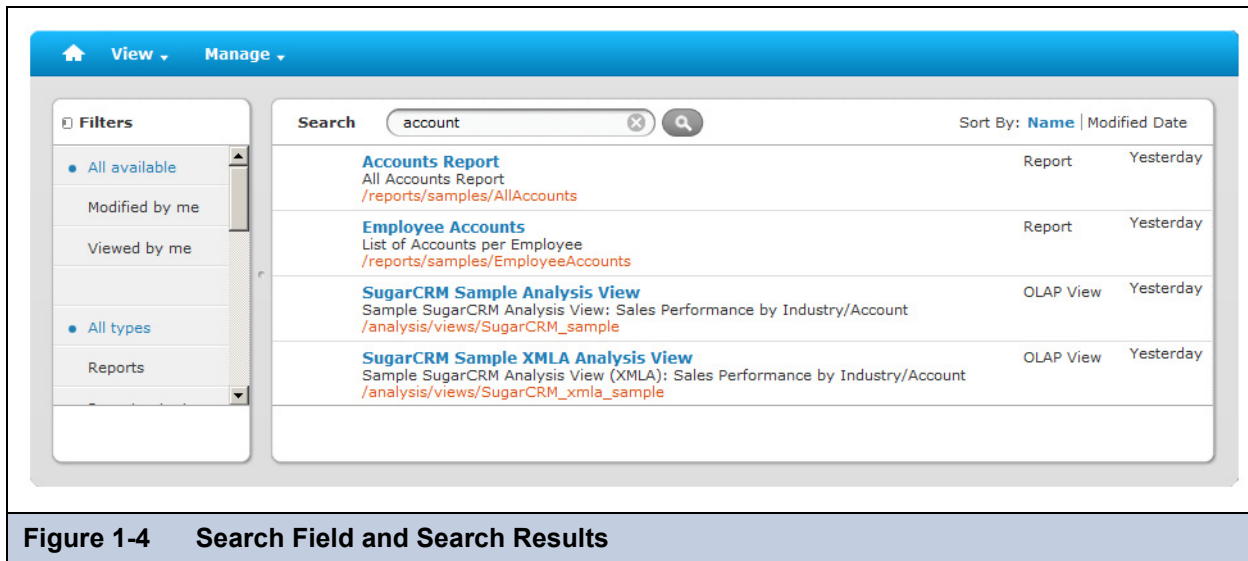
If you click **View > Search Results** and click  on the search results page, the server uses the filters you set in the **Filters** panel.

Figure 1-4 shows the results of a search for the term “account” using the filters All available and All types.



**Figure 1-4 Search Field and Search Results**



The search term you enter in the search field isn't cleared automatically. To clear the search term, click the icon in the search field.

You refine a search using filters. For example, filters can help you find your most recently viewed reports. You can set each filter independently of the others. You can set the following types of filters:

- User
- Resource
- Access time
- Scheduled report

The user filter has the following settings:

Filter Setting	Description
All Available (default)	All resources.
Modified by me	Selects only resources that were last modified by the user who's logged in.
Viewed by me	Selects only resources that were run and viewed by the user who's logged in. This filter not only applies to visualization types, but also to resources that are included in reports such as images.

The resource type filter has the following settings:

Filter Setting	Description
All types (default)	All resources.
Reports	Displays only reports.
Report outputs	Displays only the output from reports that were scheduled or run in the background. Report output can be any of the supported export types, such as HTML and PDF.
OLAP views	Displays only analysis views (if you implement Jaspersoft OLAP).
Data sources	Displays only data sources.

The access time filter has the following settings. All times are relative to the user's effective timezone:

Filter Setting	Description
Any time (default)	All resources.
Today	Resources viewed or modified since the previous midnight.
Yesterday	Resources viewed or modified during the previous day ending at midnight.
Past week	Resources viewed or modified during the past 7 days, including today.
Past month	Resources viewed or modified during the past 30 days, including today.

The scheduled report filter has the following settings:

Filter Setting	Description
Any schedule (default)	All resources.
Scheduled	Only reports that have scheduled jobs.
Scheduled by me	Only reports that have jobs scheduled by the currently logged in user.
Not scheduled	Only reports that don't have scheduled jobs and all other resource types.

Remember these do's and don'ts when searching for resources:

- Do use word fragments.
- Do search for the display name or part of the display name of a resource.
- Do search for words or fragments in the description of a resource.
- Do use multiple words.
- Don't search for folder names.
- Don't enter quotes around terms or symbols between terms.
- Don't worry about using upper- or lower-case letters in search terms.

## 1.4 Using Repository Resources

After finding a resource in the repository, naturally you want to do something with it. Options are:

- Click the name of a report to run and view it.
- Right-click the name of a resource to access other operations on the context menu, for example **Edit**. The menu items that you see depend on your access permissions.
- Click anywhere in the row except the resource name to select a resource. Ctrl-click anywhere in the rows to select multiple resources. Use the context menu or buttons above the results list: **Run**, **Edit**, **Open**, **Copy**, **Cut** (move), or **Delete**. If the button is unavailable, the resource doesn't support the operation or you don't have permission to do this operation.  
You might also need permission to access the folder or dependent file, such as an image, of a resource. For example, to schedule a report, you need to have read/write/delete permission on the folder where server saves the report output.

After scheduling a report, the  icon appears beside it in the repository. Click  to view the list of jobs scheduled for the report. For more information, see section 2.3, “[Scheduling Reports](#),” on page 19.

## 1.5 Moving Folders

If you have read permission on folders and resources, you can copy and cut them from one folder and paste them to another if you have write permission on the destination folder. The server pastes all contents of the folder that you copy or cut into the new location.

You can drag-and-drop the objects instead of using the paste menu item. Move folders one at a time. You can move other resources in batches.

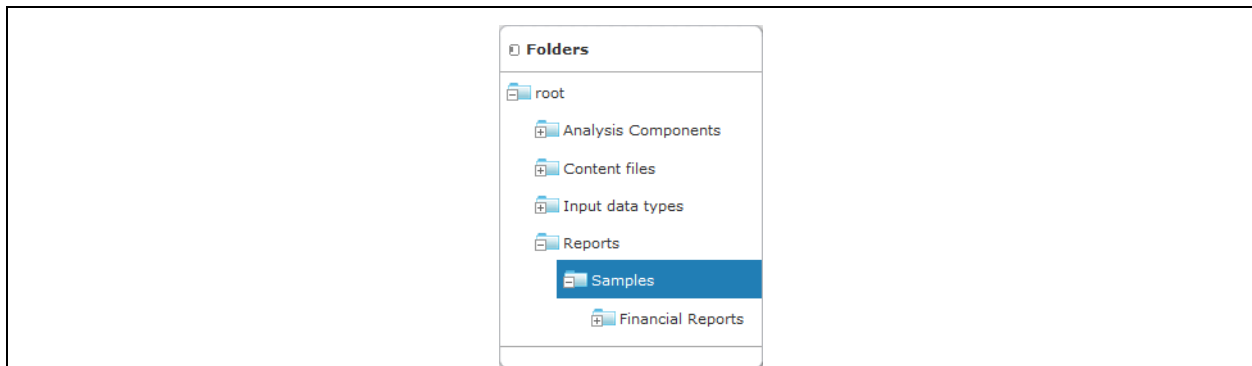


Relocated objects inherit permissions from the destination folder, losing the permissions in place prior to the move. To change permissions on an object, set the permissions explicitly.

### To move folders and resources by cutting and pasting:

1. Log into the server as a user who has these permissions:
  - ♦ Read permission on the folder or resource to move
  - ♦ Write permission on the destination folder
 For example, log in as joeuser (use the password, joeuser).
2. Click **View > Repository**.
3. In the **Folders** panel, right-click **Reports > Samples**, and select **Add Folder**.
4. In the **Add Folder** dialog, enter a name, such as Financial Reports, and click **Add**.

The Financial Reports folder appears as a subfolder of Samples and inherits Joe User's default permissions (read-write-delete) on the parent folder.



**Figure 1-5 New Financial Reports Folder**

5. The Financial Reports folder deserves a more prominent location. Move it up one level:
  - a. In **Folders**, right-click Financial Reports, and select **Cut**.
  - b. Right-click **Reports**, and select **Paste**.

The Financial Reports folder now appears in **Reports** at the same level as **Samples**.



You can relocate a folder, subject to permissions, anywhere in the repository with one exception: The server doesn't support copying and pasting a folder to the same location. If the Paste command is disabled when you right-click a destination folder, you don't have write permission on the folder.

## 1.6 Sorting the Repository List

To change the order of the list of reports and other resources, use the **Sort By** controls:

- ♦ Click **Name** to sort alphabetically (A at the top). This is the default sort order.
- ♦ Click **Modified Date** to sort by the latest modified time and date (most recent at the top).

## 1.7 Mastering Report Design

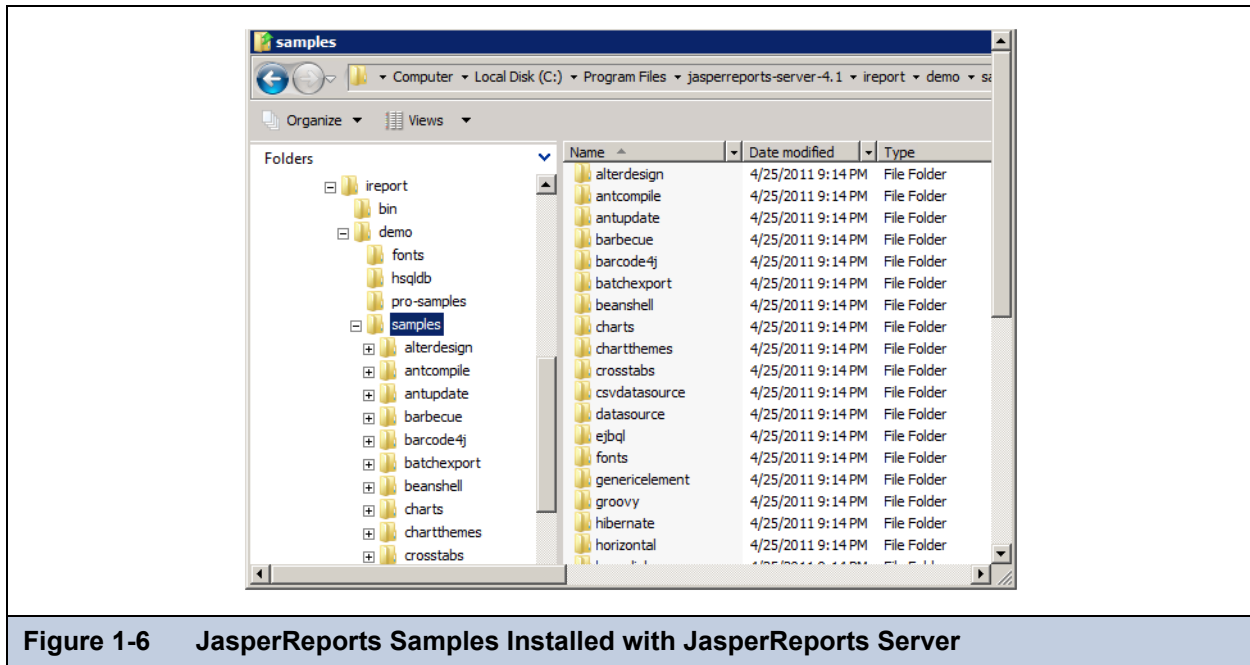
JasperReports open source library drives the reporting engine of the server. The default server installation includes iReport and an extensive set of free JasperReports samples. To master report design or to learn a little about one area, such as charting, take advantage of these samples.

### 1.7.1 Locating JasperReports Samples

**Figure 1-6** shows the location of these sample files in <js-install>\ireport\demo\samples after installing the server.



<js-install> is the root directory where JasperReports Server is installed.



**Figure 1-6 JasperReports Samples Installed with JasperReports Server**

The standalone versions of iReport and JasperReports also install the samples.

### 1.7.2 Learning about the Samples

The samples are documented online:

- [JasperReports Samples Overview](#)
- [JasperReports Samples Reference](#)

Click the Docs link on our [community website](#) to find more documentation about iReport and JasperReports Server.

## CHAPTER 2 RUNNING REPORTS

JasperReports Server makes it easy to run reports. Find a report in the repository, and simply click its name. If the report has input controls, you run the report with one set of data and then another. Using the report scheduler, you can run reports repeatedly and unattended during off hours or at other times. This chapter contains the following sections:

- **Running a Simple Report**
- **Running a Report with Input Controls or Filters**
- **Scheduling Reports**
- **Event Messages**

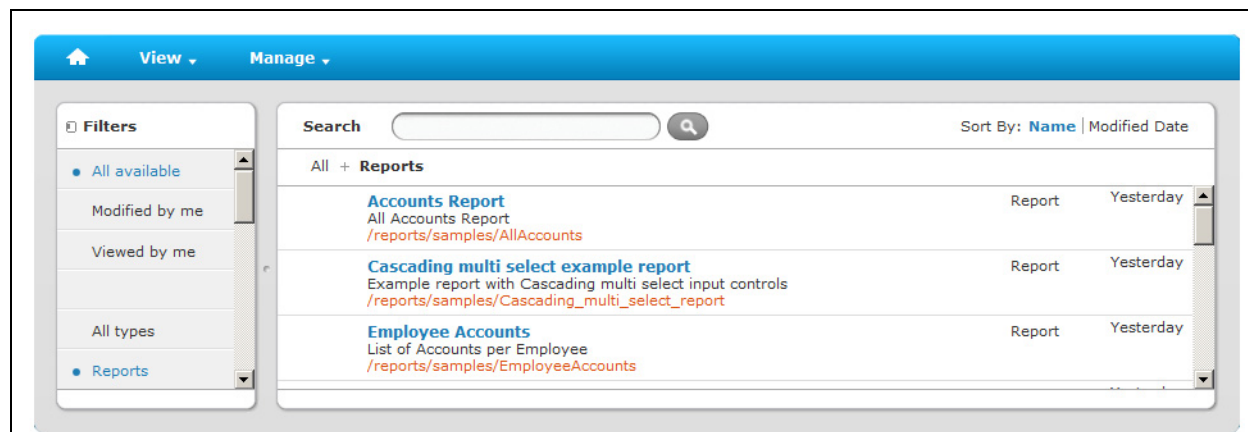
### 2.1 Running a Simple Report

This section describes how to run a tabular report that lists accounts.

#### To run a report:

1. Log into the server as an administrator, such as jasperadmin.
2. Click **View > Search Results**.
3. In **Filters**, click **Reports**.

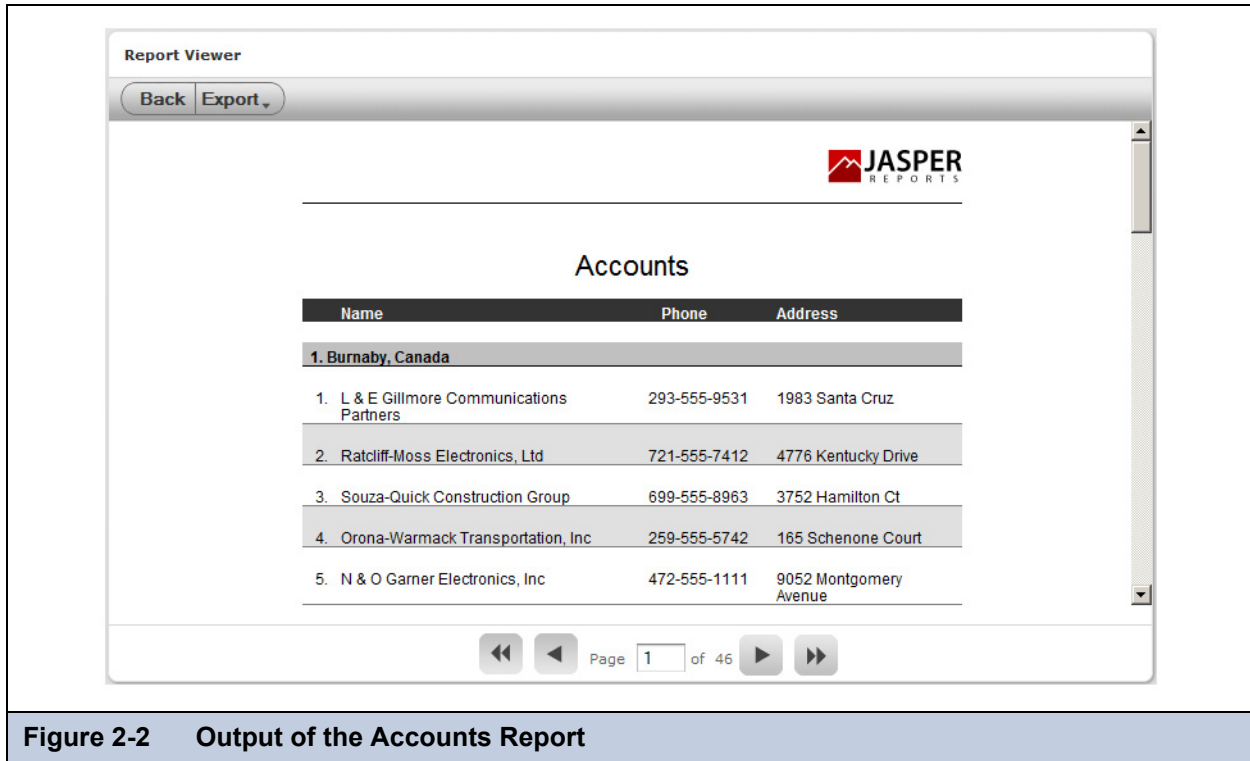
The search results appear, listing reports that your user account has permission to view.



**Figure 2-1 Search Results Listing**

Use the scroll bar to view the entire list of reports. You can view other users' reports if you have proper permissions.

- Click the name, **Accounts Report**, to run the report. Or, right-click anywhere in the report row and select **Run** from the context menu. The report appears, containing accounts and contact information.



**Figure 2-2 Output of the Accounts Report**

- Use the arrow buttons at the bottom of the report viewer to turn pages or go to the end or beginning of the report.
  - To view and save the report in other formats, click the **Export** button.
  - Select an export format:
    - PDF (Adobe Acrobat)
    - Excel (XLS)
    - CSV (Comma Separated Values)
    - DOCX (Word)
    - RTF (Rich Text Format)
    - Flash (SWF)
    - ODT (OpenDocument Text)
    - ODS (OpenDocument Spreadsheet)
    - Microsoft Open XML Format Spreadsheet (XLSX)
- The file appears in its application, for example Adobe Reader if you export to PDF.
- Save the formatted file to your hard drive from the application.
  - To return to the search results page, click the **Back** button.



If you export a report to CSV format that includes characters outside the Latin 1 character set, and the results don't look correct when opened in Microsoft Excel, try saving the file and using Excel's Import functionality.

## 2.2 Running a Report with Input Controls or Filters

The server filters the data in the report output when you run a report with an input control. The perfect input control limits the data to what you want to see—and nothing more.




## 2.2.1 Simple Input Controls

The Freight Report example has three input controls:


- Country
- Request Date
- Order ID

Using input controls, you run the report with one set of data and then another.

**To run a report with simple input controls:**

1. In the search field, enter `freight` and click .
2. In the list of search results, click the name, **Freight Report**.

The report shows data about orders numbered 10600 and above, shipped to Mexico after June 1, 1997.



**Report Viewer**

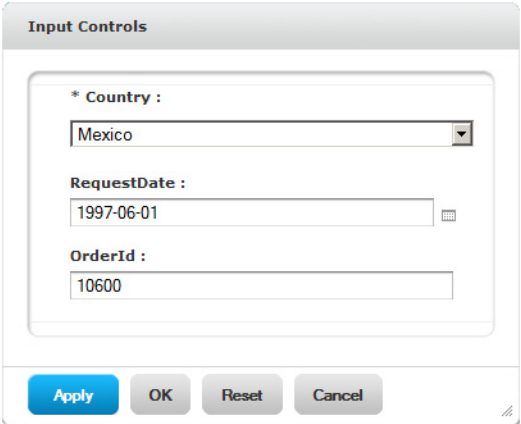
Back Options Export

*Shipping Report*

Order Id	Order Date	Required Date	Ship Country	Postal Code
10625	8/7/97 12:00 AM	04-Sep-97	Mexico	05021
10676	9/21/97 12:00 AM	19-Oct-97	Mexico	05033
10677	9/21/97 12:00 AM	19-Oct-97	Mexico	05023
10682	9/24/97 12:00 AM	22-Oct-97	Mexico	05023
10759	11/27/97 12:00 AM	25-Dec-97	Mexico	05021
10842	1/19/98 12:00 AM	16-Feb-98	Mexico	05033
10856	1/27/98 12:00 AM	24-Feb-98	Mexico	05023
10915	2/26/98 12:00 AM	26-Mar-98	Mexico	05033
10926	3/3/98 12:00 AM	31-Mar-98	Mexico	05021
10995	4/1/98 12:00 AM	28-Apr-98	Mexico	05033
11089	5/3/98 12:00 AM	31-May-98	Mexico	05033
11073	5/4/98 12:00 AM	01-Jun-98	Mexico	05033

**Figure 2-3 Freight Report Showing Mexican Data**

3. To run the report with other input option values, click **Options**. The Input Controls dialog appears.



**Input Controls**

\* Country :  
Mexico

RequestDate :  
1997-06-01

OrderId :  
10600

Apply OK Reset Cancel

**Figure 2-4 Default Input Controls**

4. Select USA in the Country field and click **OK**.



The **Apply** button refreshes the report with the new data, but keeps the Input Controls dialog open.

- In Input Controls, click **OK**.

The report shows data for the USA instead of Mexico.




**Figure 2-5 Freight Report Showing USA Data**

## 2.2.2 Cascading Input Controls

Cascading input controls in a report reduce a large number of choices to a manageable number. A single value chosen for a cascading input control determines which other values appear as choices for input. For example, the choice of a country determines which states or regions are listed as choices. For more information, see section 3.4, “[Adding Cascading Input Controls to a Report](#),” on page 54.

**To run a report with cascading input controls:**

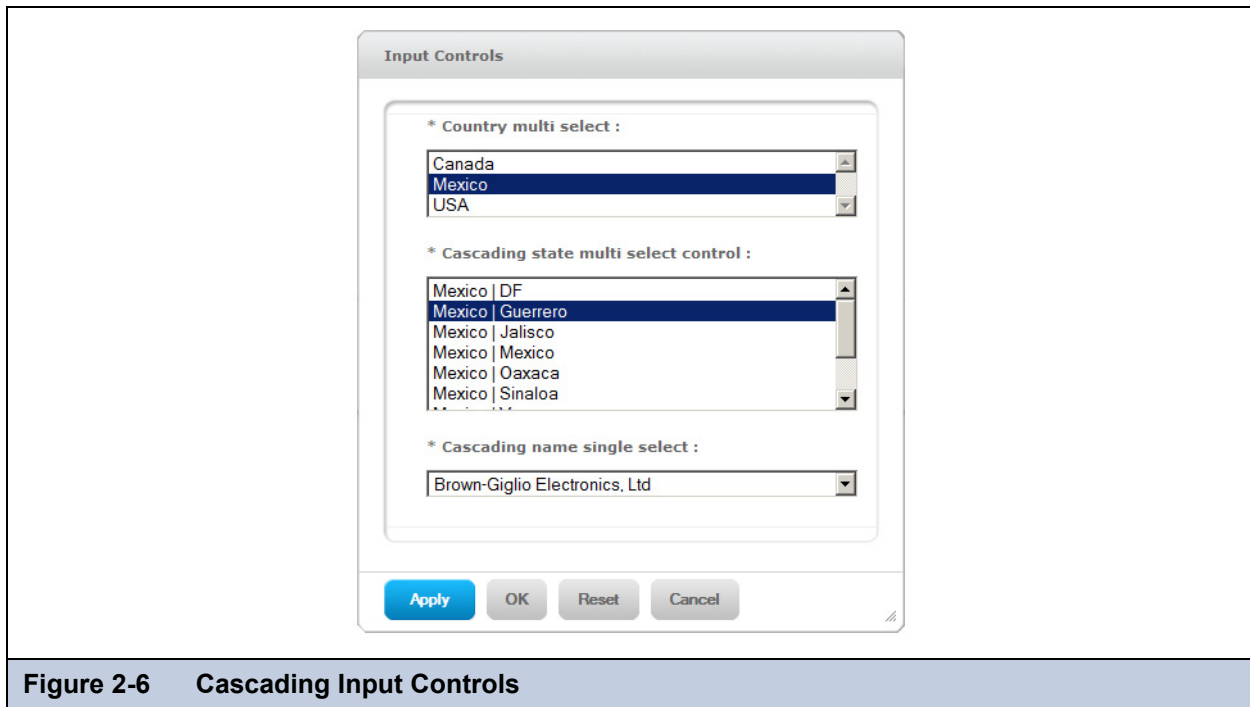
- In the **Search** field at the top of any JasperReports Server page, enter `cascading` and click  .  
The search results are displayed in the repository.
- In the list of search results, click the name Cascading multi select example report.  
Instead of running the report, you’re prompted to enter new values.
- In the **Country multi select** drop-down, select a different country, for example Mexico.  
The other drop-downs in the Input Controls dialog are automatically updated with Mexican data.



Cascading input controls are implemented as queries that access the database to retrieve the new values. The server displays an activity monitor while the query is running, and in the case of long queries, you can click Cancel and select different values.

- In the **Cascading state multi select control** drop-down, select a different state, for example Guerrero.

The **Cascading name single select** drop-down is updated to list accounts in Guerrero.



**Figure 2-6 Cascading Input Controls**



The report design defines country and state input controls as multi-select; users can Ctrl-click to select multiple values in this list.

5. Click OK to run the report with the chosen values.

The report creates a list of all account names in the chosen states. The report lists the account name chosen in the **Cascading name single select** input control at the top.

## 2.3 Scheduling Reports

Using the report scheduler wizard, you set up a job, report parameters, and output options:

- Set up a job – A job name, when to run the job, and how often
- Report parameters – If the report was designed with input controls, which parameters the job will use
- Output options – Where to store the output, in which formats, the locale, and email options for sending the output as an attachment

Jobs that you schedule run the reports in the background so you can use the server while the report runs.

The permissions of the user who schedules a job determine the data that the report exposes. For example, Gloria only has access to inventory data from the Southeast US region. A report that she schedules only shows data from that region, even when the report is viewed by users in other regions. Other users schedule the report themselves to see the data for their own regions.




Sensitive data could be exposed to unauthorized users if you schedule a job as an administrative user with no data restrictions because the report will contain all requested data in the data source. Any user who receives the report can view all the data regardless of the user's access restrictions.

## 2.3.1 Setting Up a Job

To set up a job:

1. Click **View > Reports**.
2. Use the search field or browse the list of reports to find the report you want to schedule.




If the report already has a schedule that you want to add, modify, or delete, click  beside the report name.

3. Right-click the report and select **Schedule** from the context menu.

The Scheduled Jobs page appears, as shown in [Figure 2-9 on page 23](#).

4. Click **Schedule Job**.


The Set Up the Job page of the Scheduler wizard appears.

5. In the **Job Name** field, enter a name for the job, for example, Weekly Report. The description is optional.
6. Set these schedule attributes:
  - ♦ Start – Immediately or in the future. Click  to set a future start date and time. If you don't specify recurrence, the start date and time determines when the report runs. If you specify recurrence, you may still want to delay the first report date, for example a weekday report that you want to start next week.
  - ♦ Time Zone – Specify the time zone for the schedule. The default time zone is the time zone of the server, the time zone you entered at log in. If you're in a different time zone, set this field accordingly.
  - ♦ Recurrence – Choose one of these settings:
    - ♦ None – Run the report once.
    - ♦ Simple – Schedule the job to recur at an hourly, daily, or weekly interval.
    - ♦ Calendar – Schedule the job to recur on days of the week or days of the month.

[Figure 2-7](#) shows a job set to run immediately, no recurrence:

**Figure 2-7 Set Up the Job Page**

7. Schedule the job to start at a specific time:
  - a. In Start Job, select On.

- b. Click  .  
The calendar appears.
- c. Click a date.
- d. To set the time for the job to run, at the bottom of the calendar, repeatedly click the hours and minutes. To close the calendar, click a blank spot on the Scheduler page.  
The selected date for starting the job appears in the calendar field.
- e. In Recurrence, accept the default setting, None to run the report only once.



If you select Simple or Calendar Recurrence, additional controls appear on the page. For more information, see section 2.3.6, “Running a Job Repeatedly,” on page 24.

8. Click **Next**.  
If the report you’re scheduling has input controls that prompt for user input, the Set the Parameter Values page appears.
9. Click **Next**.  
The Output Settings page appears, as shown in [Figure 2-8 on page 22](#).
10. Set output options, as described in the next section, and click Submit to schedule the job.

## 2.3.2 Setting Output Options

On the Output Details page, you can change these settings:

- Base Output File Name – The name of the file as it appears in the repository.
- Output Description – The optional description of the file that appears to users who view the repository.
- Output Format – The available output formats. Select one, or accept the default format, PDF. When you select more than one, each format is stored as a separate file in the repository and attached as a separate file to the email notification.
- Output Locale – The locale settings for generating the report.
- To – One or more email addresses separated by commas for sending email notification.



By default, no mail server is configured in a JasperReports Server installation. The administrator must configure the server to send notifications, as described in the *JasperReports Server Installation Guide*.

- Subject – The subject line of the notification email.
- Message – Content of the notification email.
- Attach Files – A check box option that, if checked, sends the output of the report with the email notification.



Be careful when sending reports containing sensitive data by email.

- Skip Empty Reports – A check box option that, if checked, prevents the server from attaching empty report output files to email notification when report execution fails.
- Output Location – The folder for saving the report output. This field is required and you must have write permission to the folder. If you email the report and do not want to save it, specify the Temp folder.
- Sequential File Names – A timestamp added as a suffix to output files. Useful for the output of recurring jobs or for time-sensitive reports where the output must be dated. When the timestamp is used, the output filename is <basename>-<timestamp>.<extension>.
- Timestamp Pattern – An optional pattern for the timestamp, based on the `java.text.SimpleDateFormat`. Valid timestamps contain letters, numbers, dashes, underscores, and periods. The default pattern is `yyyyMMddHHmm`, for example `200906150601`.

For more information about the valid timestamp patterns, refer to:

<http://download.oracle.com/javase/6/docs/api/java/text/SimpleDateFormat.html>


- ♦ **Overwrite Files** – A check box option that, if checked, overwrites old output files with newer ones of the same name. Useful when you do not have sequential filenames, or you specify a timestamp pattern that may lead to identical filenames.

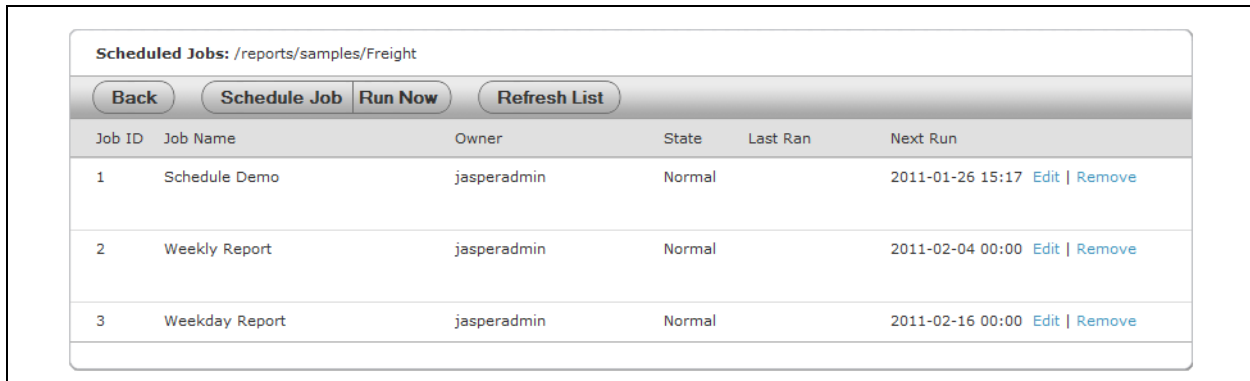
The screenshot shows the 'Scheduler' application window with the 'Output Settings' tab selected. The page is titled 'Output Settings' and includes a sub-header: 'Identify, select the format and pick a repository location for the output from this job.' The 'Job for:' field is set to '/reports/samples/Freight'. The left sidebar shows 'Set Up' and 'Parameters' with 'Output' selected. The main content area is divided into several sections: 'Output Identification' with fields for 'Base Output File Name' (containing 'Freight') and 'Output Description'; 'Output Location' with a field for 'The file will be added to:' (containing '/reports/samples') and a 'Browse...' button; 'Output Format' with checkboxes for PDF (checked), HTML, Excel, CSV, DOCX, RTF, ODT, ODS, and XLSX; 'Output Locale' with a dropdown menu set to '(Default)'; 'Email Notification' with fields for 'To:', 'Subject:', and 'Message:', and checkboxes for 'Attach Files' and 'Skip Empty Reports'; and 'Sequential File Names' with a checkbox and a 'Timestamp Pattern' field. At the bottom, there are 'Previous', 'Submit', and 'Cancel' buttons.

**Figure 2-8** Output Page for Scheduling a Report

When you click **Submit**, the job appears in the list of scheduled jobs.

### 2.3.3 Viewing the List of Scheduled Jobs

Scheduled jobs appear in the repository with a  icon beside the report name. To view the list of scheduled jobs for a report, locate a report in the repository, right-click the report, and select **Schedule** from the context menu. The Scheduled Jobs page appears for the report.



Scheduled Jobs: /reports/samples/Freight					
<a href="#">Back</a> <a href="#">Schedule Job</a> <a href="#">Run Now</a> <a href="#">Refresh List</a>					
Job ID	Job Name	Owner	State	Last Ran	Next Run
1	Schedule Demo	jasperadmin	Normal		2011-01-26 15:17 <a href="#">Edit</a>   <a href="#">Remove</a>
2	Weekly Report	jasperadmin	Normal		2011-02-04 00:00 <a href="#">Edit</a>   <a href="#">Remove</a>
3	Weekday Report	jasperadmin	Normal		2011-02-16 00:00 <a href="#">Edit</a>   <a href="#">Remove</a>

**Figure 2-9 The Scheduled Jobs Page for the Freight Report**

Typical users only see the jobs that they have defined themselves; administrators see the jobs defined by all users. In [Figure 2-9](#), Jasperadmin has scheduled three jobs for the Freight Report.

The Scheduled Jobs page shows the internal ID number of the job, the user (owner) who created the job, and the state of the job. Job states are:

- Normal – The job is scheduled.
- Running – The server is generating the report output.
- Complete – The server has finished running the job and placed output to the repository.
- Error – The scheduler encountered an error while scheduling or triggering the job. This doesn't include cases where the job is successfully triggered, but an error occurs while it runs.

The Scheduled Jobs page includes two controls:

- Edit – Changes the schedule.
- Remove – Cancels the scheduled job.

When the server receives a request to delete a job that is running, the server completes running the job before deleting it.

Buttons on the Scheduled Jobs page include:

Button	Description
<b>Back</b>	Returns to the list of reports.
<b>Schedule Job</b>	Opens the Job Details page to define a new job.
<b>Run Now</b>	Runs the report in the background, allowing you to continue working in the server. See <a href="#">section 2.3.7, “Running a Job in the Background,” on page 26</a> .
<b>Refresh List</b>	Refreshes the list of jobs, for example to see if a job has finished running.

### 2.3.4 Changing Job Schedules

When the start date for a job has passed, start a new job rather than changing the start date.

**To edit a scheduled job:**

1. Open the Scheduled Jobs page for the report, as described in [section 2.3.3, “Viewing the List of Scheduled Jobs,” on page 23](#).
2. Click **Edit** in the row of the job you want to change.

3. Make the changes on the Job Details, Schedule, Parameters, and Output pages.
4. Click **Save**. The update occurs immediately.

### 2.3.5 Stopping a Job from Running

To stop a job from running, delete it.

#### To delete a scheduled job:

1. Open the Scheduled Jobs page for the report, as described in section 2.3.3, “Viewing the List of Scheduled Jobs,” on page 23.
2. In the row of the job you want to delete, click **Remove**.

### 2.3.6 Running a Job Repeatedly

To run reports automatically on a regular basis, select simple or calendar recurrence on the Set Up the Job page:

- Simple recurrence repeatedly runs the job at an hourly, daily, or weekly interval, and is quick to set up.
- Calendar recurrence involves more settings: time of day, days of the week, or days of the month, and months of the year.

Figure 2-10 shows an example of how to set simple recurrence.


The screenshot shows the 'Scheduler' window with the 'Set Up the Job' tab selected. The dialog is titled 'Set Up the Job' and contains the instruction 'Identify the job, set start time and, optionally, select a recurrence.' The 'Job for:' field is set to '/reports/samples/Freight'. The 'Set Up' tab is active, with 'Parameters' and 'Output' tabs also visible. The 'Job Name: (required):' field contains 'Schedule Demo 2'. The 'Description:' field is empty. The 'Start Job:' section has 'On:' selected with a date and time of '2010-10-22 16:24'. The 'Time Zone:' is set to 'America/Los\_Angeles - Pacific Standard Time'. The 'Recurrence:' section shows 'Simple' selected, with 'Repeat every: 1 weeks (required)' and 'Indefinitely' selected. The 'Next' and 'Cancel' buttons are at the bottom.

**Figure 2-10 Simple Recurrence Settings**

Simple recurrence options are:

- Repeat every – The interval between jobs, in minutes, hours, days, or weeks.
- Indefinitely – Runs until you delete the job.
- Times – Runs the specified number of times.



- ♦ Until – Runs until a calendar date is reached. Click  to select the date.



If your server recognizes Daylight Savings Time (DST), jobs scheduled using simple recurrence may seem to occur one hour later (when DST ends) or one hour earlier (when DST begins). If you want jobs to recur at the same time of day and respect DST adjustments, use calendar recurrence.

Figure 2-11 shows an example of how to set calendar recurrence.

**Scheduler**

**Set Up the Job**


Identify the job, set start time and, optionally, select a recurrence.

• **Set Up**  
Parameters  
Output

Job for: /reports/samples/Freight

Job Name: (required):  
Schedule Demo 2

Description:

Start Job:  
☐ Immediately  
☒ On:  
 2010-10-22 16:24 

Time Zone: America/Los\_Angeles - Pacific Standard Time ▼


Recurrence: None | Simple | **Calendar**

Months  
☒ Every Month  
☐ Selected Months:  
 Jan  
Feb  
Mar  
Apr  
May  
Jun  
Jul  
Aug  
Sep  
Oct  
Nov  
Dec

Days  
☒ Every Day  
☐ Week Days:  
 Mon  
Tue  
Wed  
Thu  
Fri  
Sat  
Sun

☐ Month Days:

Times  
 0 Hours (required)  
 Hint: Enter 24-hour times like 9,12,15 or ranges like 9-12,1-17  
 0 Minutes (required)  
 Hint: Enter 0,15,30,45 to run every 1/2 hour


Recur Until: 

Next Cancel

**Figure 2-11 Calendar Recurrence Settings**

Calendar recurrence options are:

- ♦ Months – The months during which the report runs.
- ♦ Days – The days when the report runs:
  - ♦ Every Day
  - ♦ Week Days
  - ♦ Month Days – Enter dates or date ranges separated by commas, for example: 1, 15.

- **Times** – The time of day in minutes and hours when the job should run. The hours use 24-hour format.  
You can also enter multiple minutes or hours, and ranges, separated by commas. For example, entering 0, 15, 30, 45 for the minutes, and 9-17 for the hours, runs the report every 15 minutes from 9:00 a.m. to 5:45 p.m. Enter an asterisk (\*) to run the job every minute or every hour.
- **Recur Until** – Calendar recurrence runs until a calendar date arrives. Click  to select the date.

### 2.3.7 Running a Job in the Background

Running a job in the background generates a report, potentially long-running, without affecting performance. You can keep working in the server as the job runs. When the job completes, you can export the report directly to any format and save it in the repository. You can share a report with others by sending the generated report by email.

Running a job in the background is equivalent to scheduling the report to run immediately without recurrence.

#### To run a job in the background:

1. Click **View > Reports**.
2. Use the search field or **Filters** to find the report you want to run.
3. Right-click the report and select **Run in Background** from the context menu.
4. If the report you're running has input controls or filters that prompt for user input, a page appears for setting parameter values. Set the fields.
5. Click **Next**.  
The Output page appears. Enter the information as described in section 2.3.2, “Setting Output Options,” on page 21.
6. Click **Save**.

The report begins to run immediately.

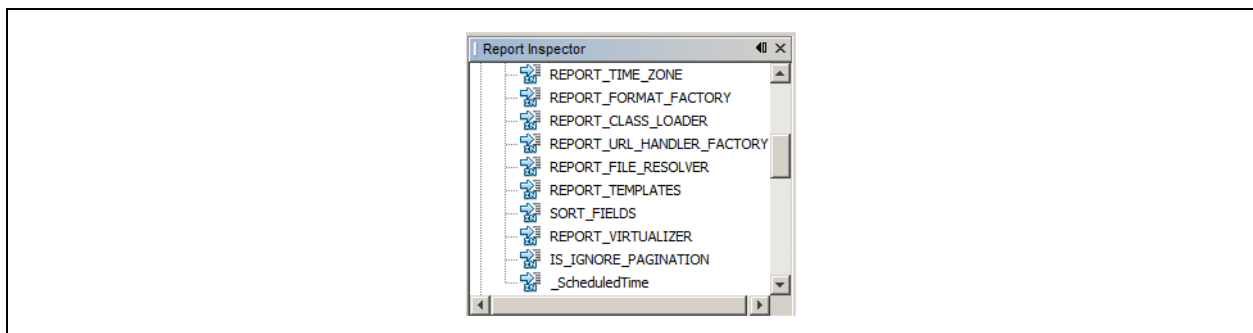
### 2.3.8 Adding a Date/Time Stamp to Scheduled Output

When you add a parameter named `_ScheduledTime` to a JRXML report design in iReport Designer, and then schedule the report to run in the server, the output includes a date/time stamp showing when the report ran. The following procedure describes how to set up and use this parameter.

#### To display the date/time that the report ran:

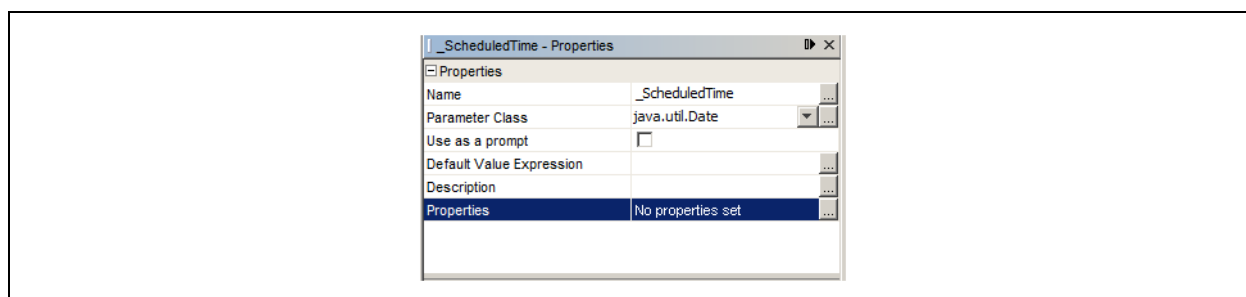
1. Launch iReport, and open an existing report.
2. In the Report Inspector, right-click Parameters, and select Add Parameter.
3. Rename the parameter `_ScheduledTime`.

The new parameter appears in the report inspector.



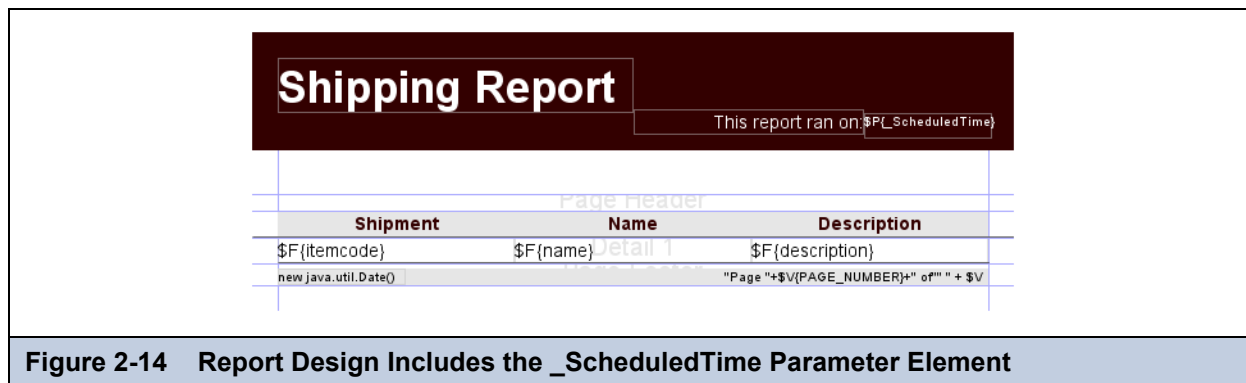
**Figure 2-12** `_ScheduledTime` Parameter in the Report Inspector

4. Set the following parameter properties:
  - Parameter Class = `java.util.Date`
  - Use as a prompt = unchecked



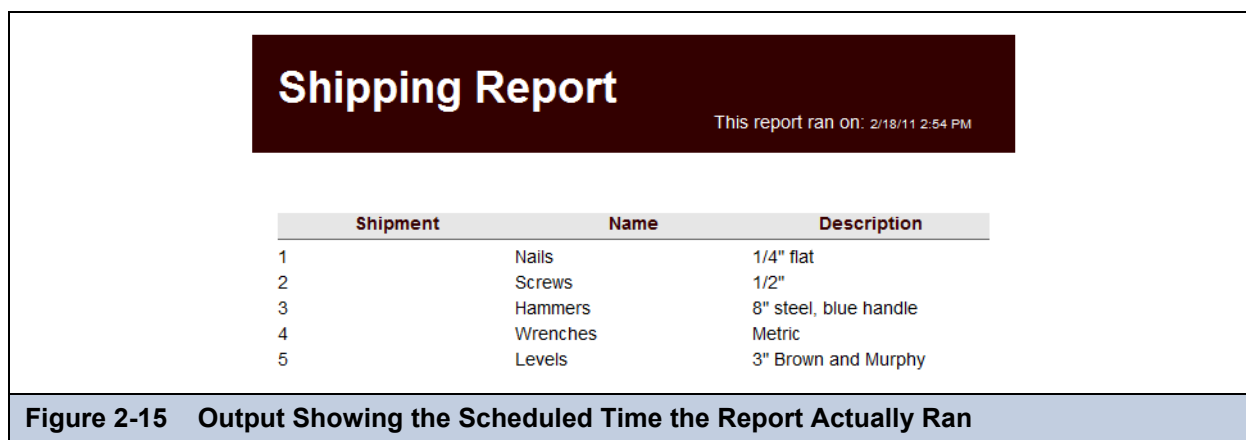
**Figure 2-13** \_ScheduledTime Parameter Properties

5. Drag the **\_ScheduledTime** element from the Report Inspector to a valid location, such as the header, in the Designer.



**Figure 2-14** Report Design Includes the \_ScheduledTime Parameter Element

6. Now you can set other properties, such as the text color of the date/time stamp. In Properties, check **Blank when Null** to prevent the word null from appearing on the report when it runs unscheduled.
7. Compile the report, and add it to the server as a report unit. For more information about how to add a report unit to the server, see section 3.2, “Adding a Simple Report Unit to the Server,” on page 30.
8. In the server, schedule the report to run immediately.
9. Open the output file.



**Figure 2-15** Output Showing the Scheduled Time the Report Actually Ran

The date and time the report actually ran appears in the output.

## 2.4 Event Messages

When an event occurs (for example, a scheduled report returns errors), JasperReports Server sends the owner of the report a notification message. You can browse these messages to troubleshoot report scheduling problems in the server. For example, you can determine that a report fails because its data source configuration uses incorrect credentials.



A common cause of the error message indicating that the report failed to execute is a misconfigured mail server. The mail server must be manually configured after installation in order for users to send email notifications.

The Messages page displays the list of events logged for the current user.

**To open the Messages page:**

1. Click **View > Messages**.

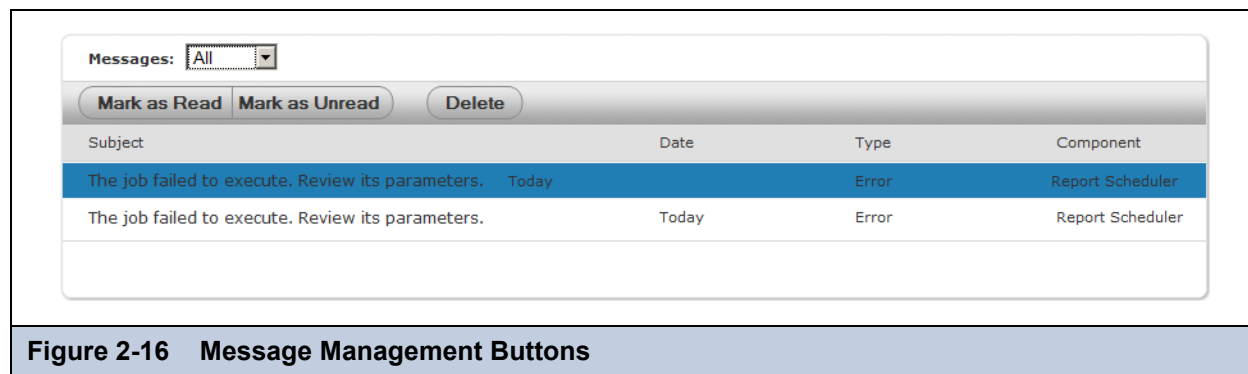
The Messages page appears.

2. To view a message, click its name.

The message opens in the Event Details page.

3. To activate the buttons on the Messages page, click in a blank area of the message row that you want to manage.

The buttons appear.



**Figure 2-16 Message Management Buttons**

4. Use the buttons on the Messages page to manage the list of messages.

## CHAPTER 3    ADDING REPORTS DIRECTLY TO THE REPOSITORY

---

To add a report to the repository, you need to have a valid JRXML file. To create this file, you can use iReport Designer. Jaspersoft recommends iReport for most users because its graphical user interface simplifies the job. You can also use a text editor to create the file containing JRXML code if you have a thorough understanding of the JasperReports file structure.

You can add a report to the server's repository in two ways:

- From within the server  
Add a JRXML file and any other resources the report needs as a report unit. A wizard guides you through each step.
- From iReport Designer  
Design the report in iReport, and use the JasperReports Server Plug-in to add the JRXML and resources to the repository.

This chapter includes examples of adding a report to the repository using the server's wizard and the plug-in. The chapter contains the following sections:

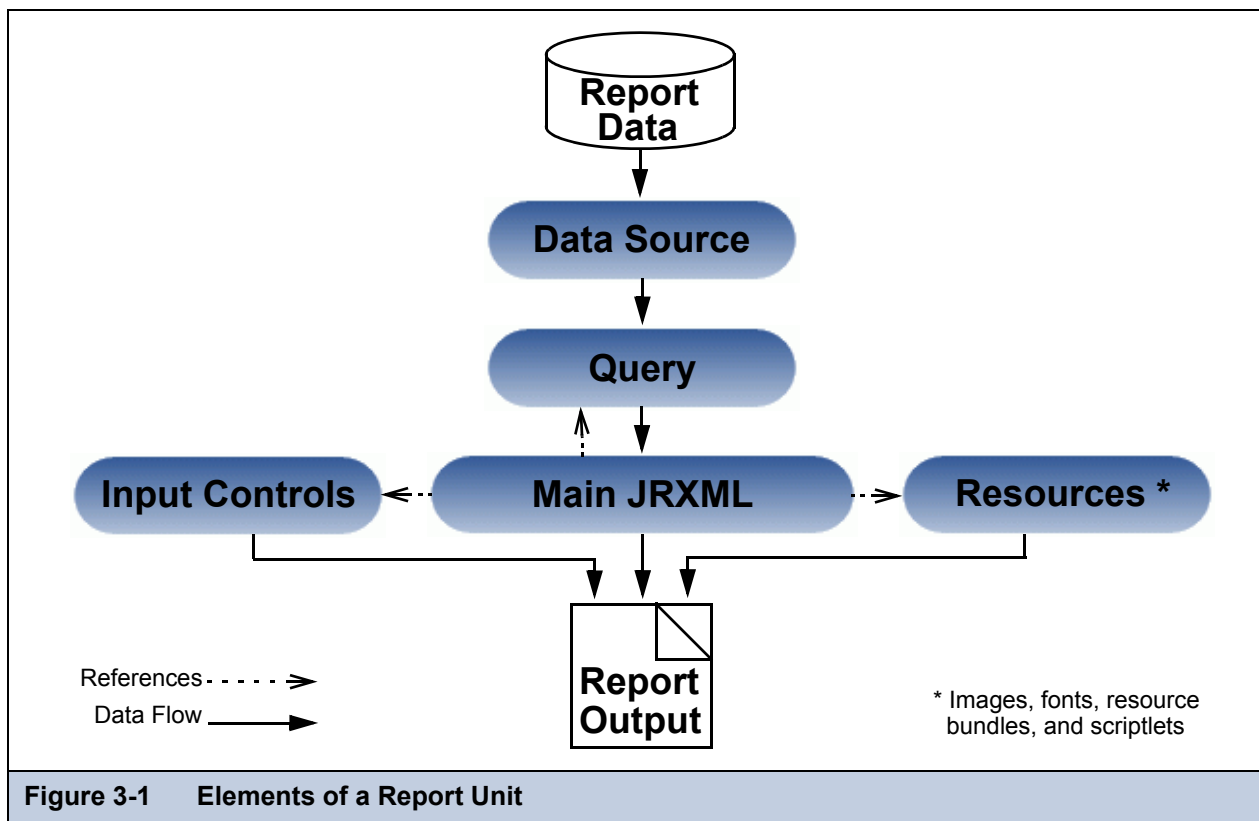
- **Overview of a Report Unit**
- **Adding a Simple Report Unit to the Server**
- **Adding a Complex Report Unit to the Server**
- **Adding Cascading Input Controls to a Report**
- **Editing JRXML Report Units**
- **Localizing Reports**

### 3.1    Overview of a Report Unit

In the server, a report unit is the collection of elements for retrieving data and formatting output. **Figure 3-1 on page 30** shows these elements:

- The data source and query that retrieves data for the report.
- The main JRXML that determines the layout and is the core of the report unit.
- The main JRXML defines other elements in one of the following ways:
  - Creating definitions internally
  - Referring to existing elements in the repository using the `repo:` syntax
- The input controls and other resources.

For more information about the report unit, refer to the *JasperReports Server Ultimate Guide*.



## 3.2 Adding a Simple Report Unit to the Server

This section presents an example of uploading a JRXML to the server. The report only has two image resources. The example defines a custom query for the report unit.

To add the simple report unit to the server, you need access to the following sample data installed with the server:

- <js-install>/samples/reports/AllAccounts.jrxml
- <js-install>/samples/images/logo.jpg files

### 3.2.1 Uploading the Main JRXML

To upload a report unit to the server, first set up a name in the repository. Next, upload the main JRXML file that references all other elements, as described in this procedure.

**To upload the main JRXML for the simple report example:**

1. Log into the server as administrator and select **View > Repository**.



If you log in as a user, you can upload a report unit to the server, but this example requires an administrator login to access the image resources.

2. Locate the folder where you want to add the report.  
In this example, go to **root > Reports > Samples**.
3. Right-click the Samples folder and select **Add Resource > JasperReport** from the context menu.



**Add Resource** only appears on the context menu if your user account has write permission to the folder.

The Set Up the Report page of the JasperReport wizard appears.

4. In **Naming**, enter the name and description of the new report, and accept the Resource ID generated as you type the name:
  - ♦ **Name**  
Display name of the report: `New Simple Report`
  - ♦ **Resource ID**  
Permanent designation of the report object in the repository: `New_Simple_Report`
  - ♦ **Description**  
Optional description displayed in the repository: `This is a simple example`
5. Select **Upload a Local File** and **Browse** to `<js-install>samples/reports/AllAccounts.jrxml`.



You can upload a new JRXML or select a JRXML from the repository.

Figure 3-2 shows the completed Set Up the Report page.

**Add JasperReport**

**Set Up the Report**

Set the required values for the report, then, optionally, proceed to other pages in the flow.

**Set Up**  
Controls & Resources  
Data Source  
Query  
Customization

Name:  
New Simple Report

Resource ID (required):  
New\_Simple\_Report

Description:  
This is a simple example

Locate the JRXML File

☒ Upload a Local File  
C:\js-install\samples\rep Browse...

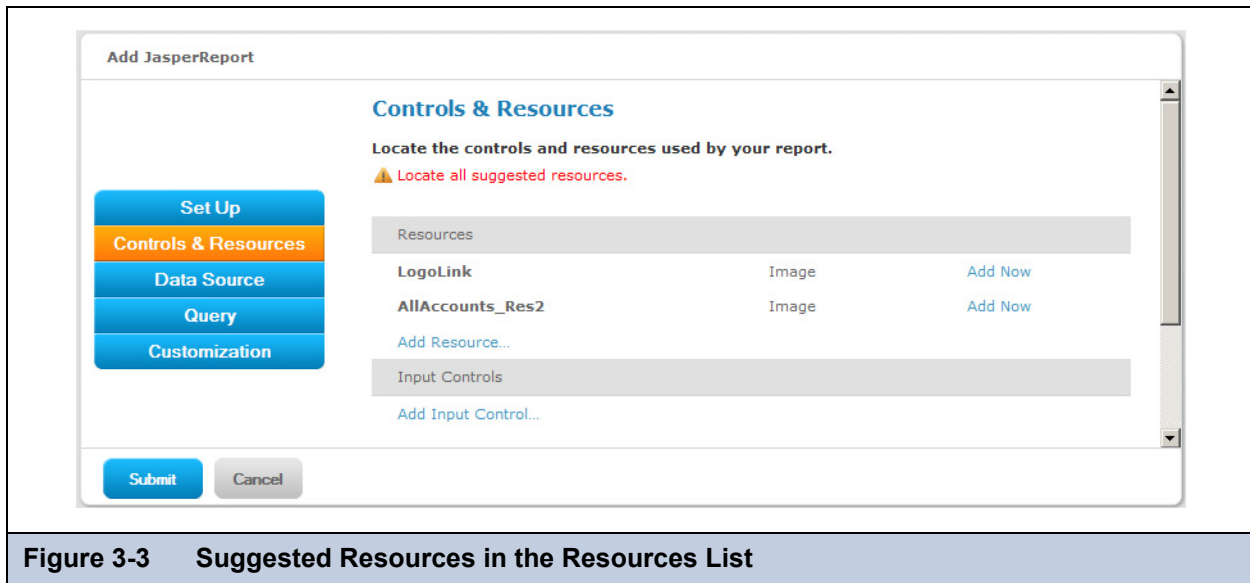
☐ Select a JRXML from the Repository  
Browse...

Submit Cancel

**Figure 3-2 Required Set Up Values**

6. Click **Controls & Resources**.

The server uploads the main JRXML and displays the list of suggested resources on the Controls & Resources page. The suggested resources are missing resources that the server needs to validate and run the report.



**Figure 3-3 Suggested Resources in the Resources List**

7. Review the hyperlinked names of images:

- ♦ Logolink
- ♦ AllAccounts\_Res2

Next, upload the suggested file resources; otherwise, the server can't validate the report.

### 3.2.2 Uploading Suggested File Resources

A JRXML file doesn't embed resources, such as images. When the server uploads the JRXML, it tries to detect missing resources and creates a list of them, as shown in [Figure 3-3](#). You need to take one of the following actions:

- ♦ Upload resources that the report needs.
- ♦ Fill the gap with resources already in the repository.

If the Controls & Resources page doesn't list any suggested file resources, perhaps the report doesn't reference any. Often, the server can't detect all of the referenced resources, as discussed in section [3.3.1, "Uploading Undetected File Resources,"](#) on [page 40](#).

On the Controls & Resources page of the JasperReport wizard, you add missing resources.

**To upload the suggested file resources for the simple report example:**

1. Click **Add Now** in the same row as LogoLink.  
The Locate File Resource page appears.
2. Choose **Select a resource from the Repository**, and **Browse** to Images/JR Logo.



Any image file works.

3. Click **Select**.



The path to the image appears in the wizard.

The screenshot shows the 'Add JasperReport' wizard at the 'Locate File Resource' step. There are two radio buttons: 'Upload a Local File' (unselected) and 'Select a resource from the Repository' (selected). Below the first option is a text box and a 'Browse...' button. Below the second option is a text box containing the path '/images/JRLogo' and a 'Browse...' button. At the bottom are 'Previous', 'Next', and 'Cancel' buttons.

**Figure 3-4 A Resource Selected from the Repository**

4. Click **Next**.

The Add a Report Resource page appears.

The screenshot shows the 'Add JasperReport' wizard at the 'Add a Report Resource' step. The title is 'Add a Report Resource' with the subtitle 'Set the properties for the resource.' Below this is a form with the following fields: 'Type: Image', 'Selected Resource: /images/JRLogo', 'Name (required): LogoLink', 'Resource ID (read-only): LogoLink', and 'Description: LogoLink'. At the bottom are 'Previous', 'Next', and 'Cancel' buttons.

**Figure 3-5 Properties of a Resource**



The properties include the LogoLink name, resource ID, and description. These properties don't redefine the properties of the JRLogo file in the repository.

5. Click **Next** to accept the default naming of the file resource.  
The Controls & Resources page appears again, showing that the LogoLink resource was added.
6. On Controls and Resources, click **Add Now** in the AllAccounts\_Res2 row.  
The Locate File Resource page appears again.

7. Select **Upload a Local File**, and **Browse** to <js-install>/samples/images/logo.jpg.

8. Open logo.jpg, and click **Next**.

The server uploads the file.

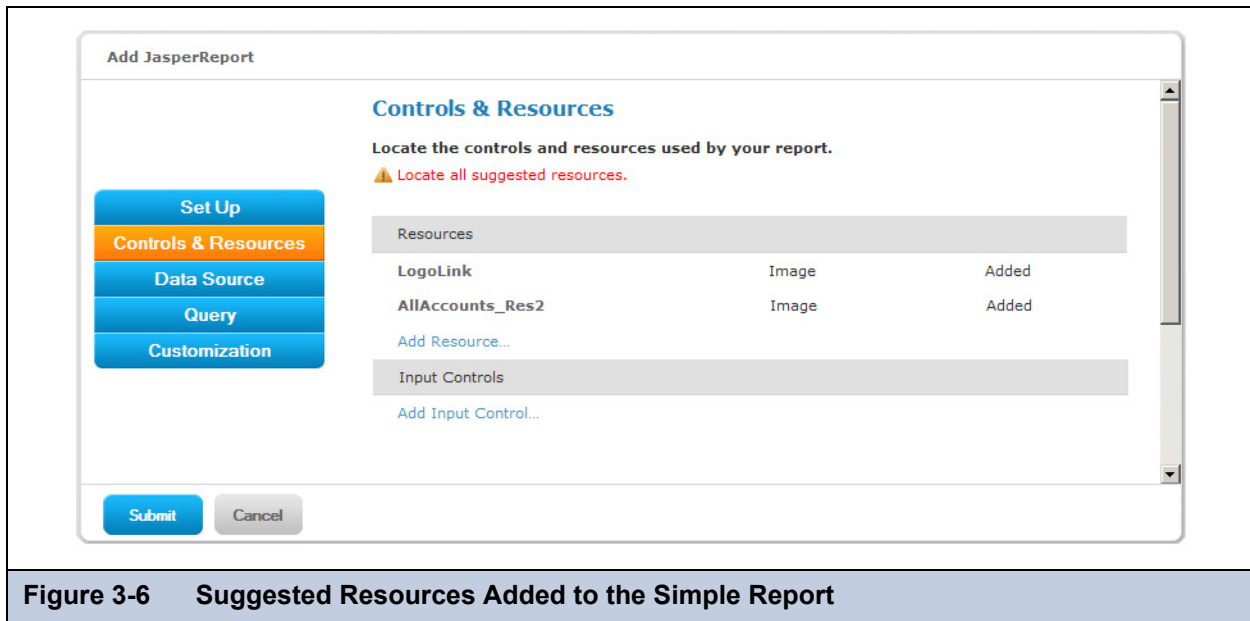
The Add a Report Resource page appears, showing the properties of the file resource.



The properties include the AllAccounts\_Res2 name, resource ID, and description.

9. Click **Next** to accept the default naming of the file resource.

The Controls & Resources page reappears, showing the addition of both resources referenced in the main JRXML.



**Figure 3-6 Suggested Resources Added to the Simple Report**

10. Click **Data Source** and define a data source as described in the next section.

### 3.2.3 Defining the Data Source

Data sources belong to the report engine, JasperReports Server, and are not defined in the main JRXML. The JRXML does not retain any data source defined in iReport when you add the JRXML to the server. You need to define a data source in the server that the report unit can use. From the Data Source page, select a data source in the repository or create a new data source on-the-fly.

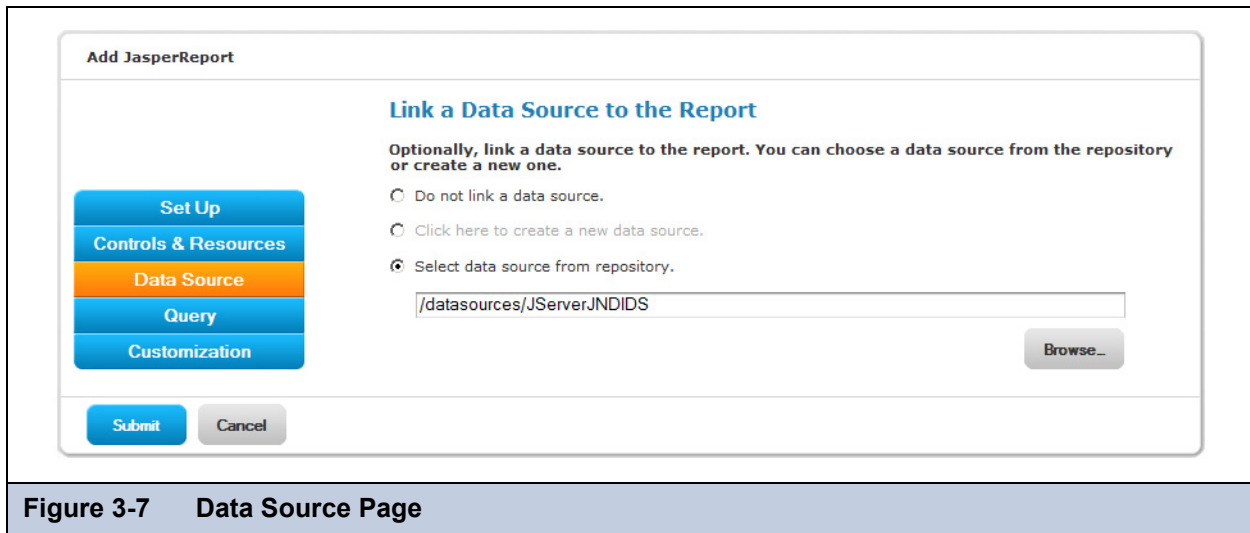
#### To define the data source for the simple report example:

1. In the JasperReport wizard, click **Data Source**.

The Link a Data Source to the Report page presents these choices:

- ♦ Do not link a data source – Select or define the data source at a later time. You see an error if you run the report in this state.
- ♦ Click here to create a new data source – Define a new data source that is only available to your report.
- ♦ Select data source from repository – Select an existing data source from the repository.

2. Select **Select data source from the Repository** and **Browse** to /Data Sources/JServer JNDI Data Source. Click Select. The path to the data source appears.



**Figure 3-7 Data Source Page**

3. Click **Query** to define to the Query as described in the next section.

### 3.2.4 Defining the Query

The query in the report unit determines the data that the server retrieves from a data source. You can use an existing query or define a new one. You can create multiple reports that look the same but contain different data by defining different queries for the same JRXML file. The simple report example defines a custom query that displays accounts from a single country.

**To define a custom query for the simple report example:**

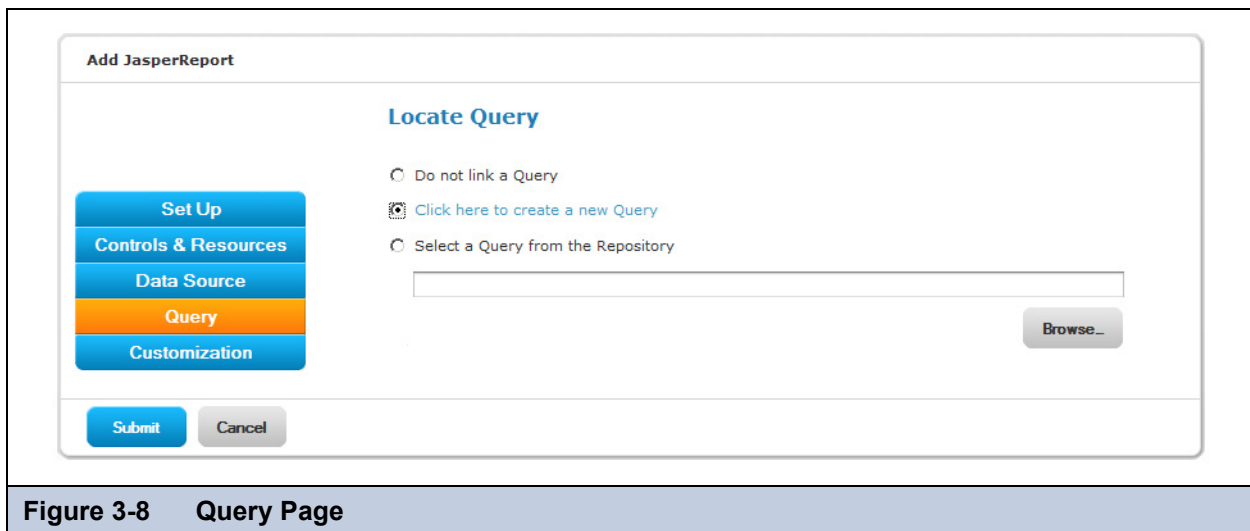
1. In the JasperReport wizard, click Query.

The Locate Query page presents these choices:

- Do not link a Query – Uses the query, if there is one, defined within the main JRXML. If the main JRXML doesn't have a query, you can't run the report.
- Click here to create a new Query – Guides you through defining a new query, only available to your report.
- Select a Query from the Repository – Selects an existing query from the repository.

The AllAccounts.jrxml file uploaded in section 3.2.1, “[Uploading the Main JRXML,](#)” on page 30 already contains a query. This example overrides the existing query by defining a new one.

2. On the Query page, select the **Click here to create a new Query** option.



**Figure 3-8 Query Page**

- Click the link, **Click here to create a new Query**.

The Name the Query page appears where you can enter the name, resource ID, and description of the query. This query and its properties are visible only within the report unit.

- In this example, the query must retrieve only Canadian accounts. Enter the following values:
  - Name – CanadaAccounts
  - Resource ID – CanadaAccounts
  - Description – Query for New Simple Report in User Guide

The screenshot shows the 'Add Query' dialog box with the 'Name the Query' step selected. The dialog has a sidebar with three options: 'Name the Query' (selected), 'Link a Data Source', and 'Define the Query'. The main area is titled 'Name the Query' and contains the instruction 'Enter the required property values.' Below this are three input fields: 'Name (required):' with the value 'CanadaAccounts', 'Resource ID (required):' with the value 'CanadaAccounts', and 'Description:' with the value 'Query for New Simple Report in User Guide'. At the bottom are three buttons: 'Previous', 'Next', and 'Cancel'.

**Figure 3-9 Name the Query Page**

- Click **Next**.

The Link a Data Source to the Query page appears. You can select a query from the repository, define a new one, or select not to link a data source.

- Select **Do not link a data source** to use the data source you selected in section 3.2.3, “Defining the Data Source,” on page 34.
- Click **Next**.

The Define the Query page appears.

- Select **SQL** in the Query Language drop-down and enter the following Query String to only retrieve Canadian accounts:
 

```
SELECT * FROM accounts WHERE billing_address_country = "Canada" ORDER BY
billing_address_city
```

The screenshot shows the 'Add Query' dialog box with the 'Define the Query' step selected. The sidebar shows 'Define the Query' as the active step. The main area is titled 'Define the Query' and contains the instruction 'Select a language and enter the query.' Below this is a 'Query Language:' dropdown menu set to 'SQL'. Underneath is a 'Query String:' text area containing the SQL query: 'SELECT \* FROM accounts WHERE billing\_address\_country = "Canada" ORDER BY billing\_address\_city'. At the bottom are three buttons: 'Previous', 'Save', and 'Cancel'.

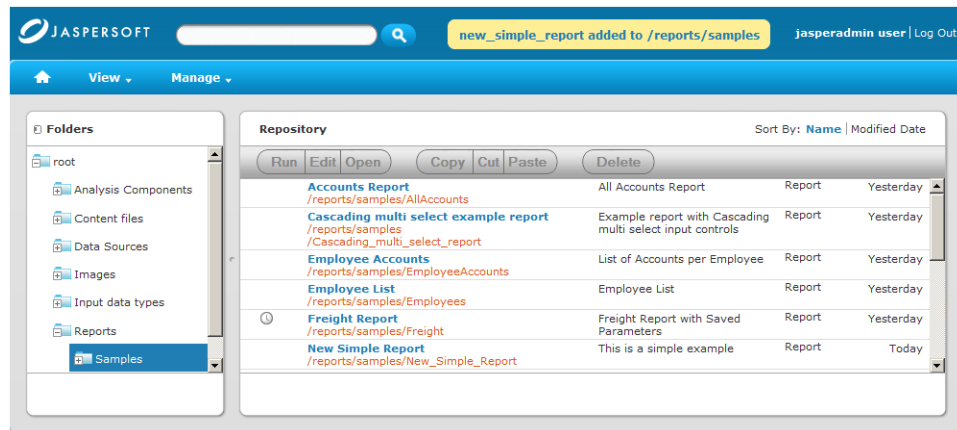
**Figure 3-10 Definition of a Query**

- Click **Save** to save the query. The Customization page appears. No customization is required.

10. Click **Submit** to submit the new report unit to the repository.

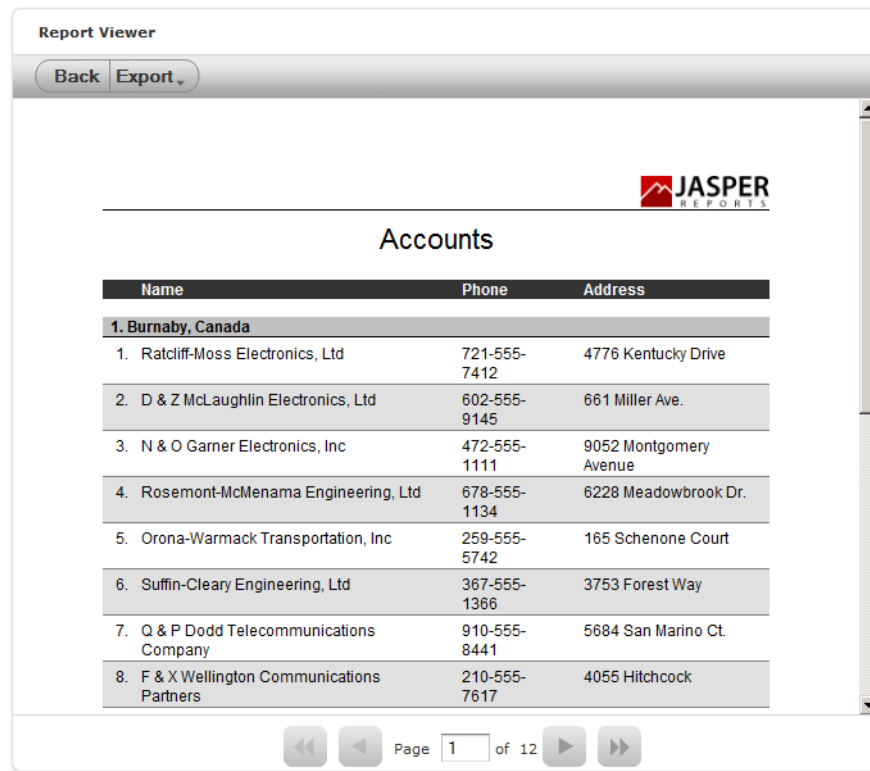
### 3.2.5 Saving the New Report Unit

To submit a new report unit to the repository, click **Submit** on any page of the JasperReport wizard, from the Set Up page to the Customization page. You don't have to set options on pages, such as Customization, you do not need. When you click **Submit**, the server validates the report unit. After clicking **Submit**, a message appears at the top of the repository page, indicating that the report was successfully saved. The report appears in the repository with the description you entered on the Set Up page. To run the report and view the output, click its name, New Simple Report.



**Figure 3-11 New Simple Report Added to the Repository**

Figure 3-12 shows the output, only Canadian accounts.



**Figure 3-12 Output of the New Simple Report**

In the Report Viewer, click  to go to the end of the report. The logo images that you added as file resources appear.



**Figure 3-13 Output of the New Simple Report Showing the Image Resources**

### 3.3 Adding a Complex Report Unit to the Server

This section includes an example of how to add a report unit with all these resources:

- SalesByMonth.jrxml file – the main JRXML
- SalesByMonthDetail.jrxml file – a subreport
- sales.properties – an English resource bundle file
- scriptlet.jar – a scriptlet class JAR file
- JR Logo – an image in the repository
- JServer JNDI data source – a data source file in the repository

These resources are part of the sample data installed with the server. To complete this example and run the report without server errors, you need access to these resources.

The example also guides you through defining every type of input control:

- Text
- Check box
- Drop-down
- Date
- Query

If you're not interested in creating all types of input controls, but want to work through part of the example, delete parameters for the input controls you don't create before you run the report. For more information, see [“To remove unwanted parameters from the sample report design:” on page 61](#).

The complex report in this example is almost exactly like the SalesByMonth report in the **Reports > Samples** folder of the repository.

#### To upload the main JRXML and suggested resource files for the complex report unit:

1. Log into JasperReports Server as administrator and select **View > Repository**.



If you log in as a user, you can upload a report unit to the server, but this example requires an administrator login to access the image resources.

2. Locate the folder where you want to add the report.

For example, navigate to **Root > Reports > Samples**.

3. Right-click the Samples folder and select **Add Resource > JasperReport** from the context menu.



**Add Resource** only appears on the menu if your user account has write privilege to the folder.

The Set Up the Report page appears.

4. On the Set Up the Report page, enter these properties:
  - ♦ Name – New Complex Report
  - ♦ Resource ID – New\_Complex\_Report
  - ♦ Description – This is a complex report
5. Select **Upload a Local File**.
6. Click **Browse** to locate the file `<js-install>/samples/reports/SalesByMonth.jrxml`.
7. Click **Controls & Resources**.

On the Controls & Resources page, the list of suggested resources appears:

- ♦ A sub-report (the SalesByMonthDetail.jrxml file)
- ♦ A logo image

**Figure 3-14** shows the list of suggested resources.



**Figure 3-14 Suggested Resources for the Complex Report**

8. On the Controls & Resources page, upload the sub-report:
  - a. Click **Add Now** in the same row as SalesByMonthDetail.  
The Locate File Resource page appears.
  - b. Select **Upload a Local File**.
  - c. Click **Browse** to locate the file `<js-install>/samples/reports/SalesByMonthDetail.jrxml`. Select SalesByMonthDetail.jrxml.
  - d. On the Locate File Resource page, click **Next**.
  - e. On the Add a Report Resource page, click **Next** to accept the default report resource name and resource ID.
9. On the Controls & Resources page, upload the logo image resource:
  - a. In the same row as Logo, click **Add Now**.  
The Locate File Resource page appears.
  - b. On the Locate File Resource page, click **Select a resource from the Repository**.
  - c. Click **Browse** to locate the file `/Images/JR Logo` and select JR Logo.

- d. Click **Next**.

The Add a Report Resource page appears.

- e. On the Add a Report Resource page, click **Next** to accept the default name, resource ID, and description: Logo.  
The second suggested resource is added.

### 3.3.1 Uploading Undetected File Resources

The JasperReport wizard can't detect every type of resource referenced in the main JRXML. You need to know the names of these resources and add them to the report; otherwise, the server can't validate the report. This document provides you with the names of these resources, but if you wanted to discover these names, use the JasperReports Server Plug-in to open the JRXML in iReport and examine its parameters and properties. For more information about the JasperReports Server Plug-in, see [Chapter 6, "Accessing Reports from iReport," on page 93](#).

These are the undetected resources in the SalesByMonth.jrxml:

- ♦ A scriptlet JAR – The scriptlet writes the message, "I'm a scriptlet in a jar," to the last page of the report output.
- ♦ An English language resource bundle.
- ♦ The optional Romanian language resource bundle.

If you're interested in working with a multi-lingual report, add the Romanian resource bundle. The Romanian resource bundle is part of the sample data installed with the server.

On the Controls & Resources page, upload the undetected resources to the server using exactly the same name for the resource ID as iReport uses.

#### To upload the undetected file resources for the complex report example:

1. Add and upload the scriptlet JAR file:
  - a. On the Controls & Resources page, click **Add Resource**.
  - b. On the Locate File Resource page, select **Upload a Local File**, and click **Browse** to locate the <js-install>/samples/jars/scriptlet.jar file.
  - c. Click **Next**.  
The Add a Report Resource page shows that uploading the file was successful. The page shows that the server recognized the type (JAR) and name (scriptlet.jar) of the selected resource. [Figure 3-15 on page 41](#) shows the Add a Report Resource page.
  - d. Enter the following information; the Resource ID is referenced in the main JRXML file, so do not change it:
    - ♦ Name – Scriptlet
    - ♦ Resource ID – Scriptlet
    - ♦ Description – Scriptlet JAR for complex report

[Figure 3-15](#) shows these values entered on the Add a Report Resource page.



**Add JasperReport**

**Add a Report Resource**

Set the properties for the resource.

Type: JAR

Selected Resource:  
scriptlet.jar

Name (required):  
Scriptlet

Resource ID (required):  
Scriptlet

Description:  
Scriptlet JAR for complex example

Previous Next Cancel

**Figure 3-15 Scriptlet JAR Resource Properties**

2. Click **Next**.
3. Add and upload the English resource bundle:
  - a. On the **Controls & Resources** page, click **Add Resource** again.
  - b. Select **Upload a Local File**, click **Browse** to locate this file:  
`<js-install>/samples/resource_bundles/sales.properties`
  - c. In Locate File Resource, click **Next**.

The Add a Report Resource page shows that uploading the file was successful. The page shows that the server recognized the Type (resource bundle) and name (sales.properties) of the selected resource.

- d. Enter this information:
  - ♦ Name – sales.properties
  - ♦ Resource ID – sales.properties



Enter the Resource ID field exactly as shown.

- ♦ Description – Default English resource bundle

4. Click **Next**.
5. Add and upload the Romanian Resource bundle:
  - a. On the Controls & Resources page, click **Add Resource** again.
  - b. Select **Upload a Local File** and click **Browse** to locate this file:  
`<js-install>/samples/resource_bundles/sales_ro.properties`
  - c. In Locate File Resource, click **Next**.

The Add a Report Resource page shows that uploading the file was successful. The server recognized the type (resource bundle) and name (sales\_ro.properties) of the selected resource.

- d. Enter the following information:
  - ♦ Name – sales\_ro.properties

- ♦ Resource ID – `sales_ro.properties`

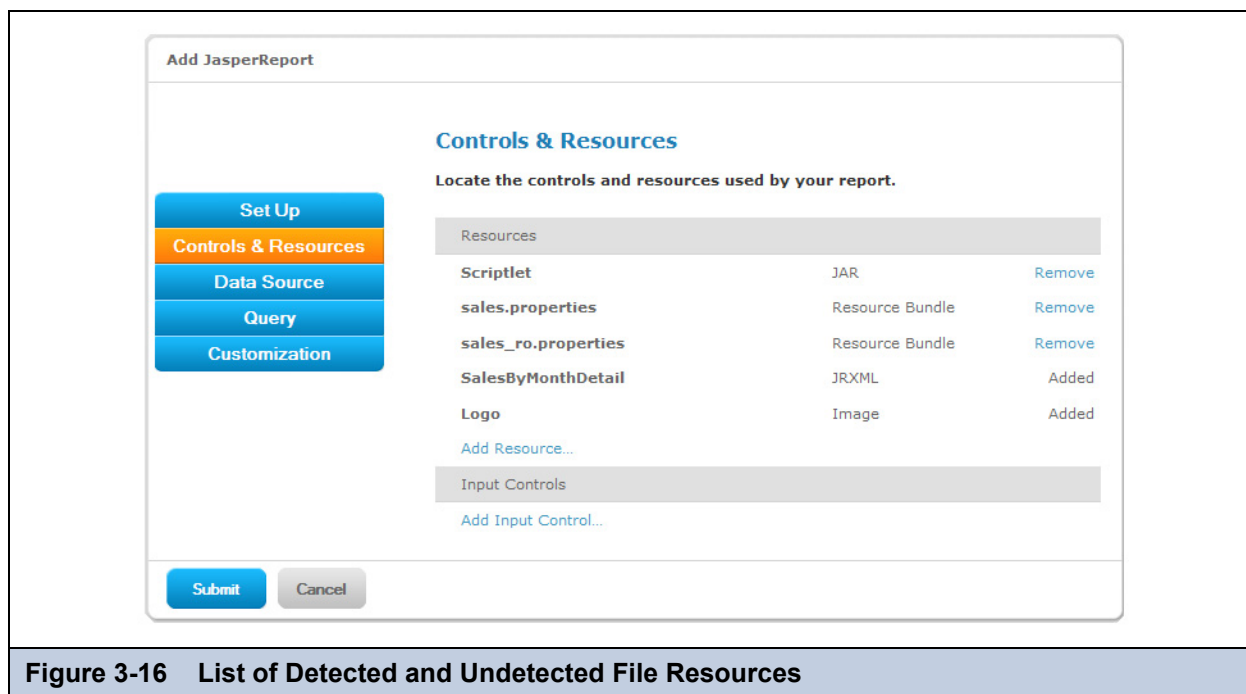


Enter the Resource ID field exactly as shown.

- ♦ Description – Romanian resource bundle

e. Click **Next**.

The Controls & Resources page lists all the files.



**Figure 3-16 List of Detected and Undetected File Resources**

If you want to upload a different file for a named resource, click its resource ID in the **Resources** list and locate the new file or repository object. You can also change the name and description of the resource, but not its resource ID. If there's a mistake in a resource ID:

- ♦ Locate the ID in the list of resources on the Controls & Resources page, and click **Remove**.
- ♦ Re-add the resource, entering the correct resource ID.

### 3.3.2 Adding Input Controls

Input controls are graphical widgets the server displays with the report. Input controls perform the following functions:

- ♦ Prompt the user for input
- ♦ Validate the format of the input
- ♦ Pass the input to the report

Based on the input, the server modifies the WHERE filter clauses in SQL parametrized queries.

Input controls correspond to the parameters defined in JRXML reports, such as `$P{name}`. The server maps the value that the user enters for the input control to the parameter of the same name. If you define an input control in the JasperReport and the server can't find a parameter by the same name in the JRXML, the input control doesn't function when the report runs.



The JRXML can define a default value for the input control. To prevent users from changing the default, don't check the input control's Visible property.

When you create an input control, you define, or use a predefined, datatype, for user entries. Datatypes define the expected input (numbers, text, date, or date/time) and can include range restrictions that the server enforces. The server uses the datatype to classify and validate the data.

To define a datatype, set properties on the Set the Datatype Kind and Properties page.

**Edit Datatype**

**Set the Datatype Kind and Properties**

First, select the kind of datatype you wish to add, then enter the required property values.

Type: Date  
Text  
Number  
Date  
Date/Time

Name (r): date\_la

Resource ID (read-only):  
date

Description:  
date description

Save Cancel

**Figure 3-17 Setting the Datatype of the Input Data**

Figure 3-17 shows the properties that define a Date datatype. Properties differ for other datatypes that appear on the Set the Datatype Kind and Properties page.

<b>Type</b>	The classification of the data: Text, Number, Date, or Date-Time.
<b>Name</b>	The name of the datatype.
<b>Resource ID</b>	The unique ID of the datatype that you can't edit.
<b>Description</b>	Any additional information you want to provide about the datatype.
<b>Pattern</b>	A regular expression that restricts the possible values of the field. Appears when the Text type.
<b>Minimum value</b>	The lowest permitted value for the field.
<b>Maximum value</b>	The highest permitted value for the field.
<b>Minimum Is Strict</b>	If checked, the maximum value itself isn't permitted; only values less than the maximum value are permitted.
<b>Maximum Is Strict</b>	If checked, the minimum value itself isn't permitted; only values greater than the minimum value are permitted.

You choose one of these widget types for the input control:

- Boolean – A check box widget for entering a yes/no value.
- Single value – A text, number, date, or date/time widget. Input can be constrained to a minimum value, maximum value, or both. Text input can also be constrained by a matching pattern. A text box widget for entering a value, or a calendar for entering date and date/time.
- Multiple values – One of the following widgets that present a static list of values, or a dynamic list of values returned by a separate query, to the user.
  - Drop-down list to select a single value
  - Radio buttons to select a single value
  - Multi-select list to select multiple values
  - Check boxes to select multiple values

After determining the list of values to be presented to the user, choose the widget.

The query in the SalesByMonth.jrxml file has several input control parameters, one for each different type of input control. These procedures show you how to add each type to the report unit.

### 3.3.2.1 Adding a Text Input Control

The simplest input control is a text box. In this example, the datatype for the input value is a number; the server verifies that the user enters a number into the text box.

**To add a text input control to the complex report example:**

1. After completing steps in **“Uploading Undetected File Resources” on page 40**, click **Controls & Resources** in the JasperReport wizard.
2. On the Controls & Resources page, click **Add Input Control**.  
The Locate Input Control page appears.
3. Select **Define an Input Control in the next step**.
4. Click **Next**.
5. On the Create Input Control page, accept the default type of input from the **Type** drop-down: **Single Value**.
6. Enter the other properties for the input control:

The name is referenced in the main JRXML file, so you must enter it exactly as shown.

- Prompt Text – The label that the user sees next to the widget for this input: `Text Input Control`
- Parameter Name – The name of the parameter in the report that receives the user value: `TextInput`
- Description – An optional description that appears only within the report wizard: leave blank in this example.
- Mandatory, Read-only, Visible – A setting that affects how the input control is displayed: check only **Visible**.

**Add Input Control**

**Create Input Control**

First, select the kind of input control you wish to add, then enter the required property values..

Type: Single Value

Prompt Text (required):  
  
The label is displayed to users next to the input control.

Parameter Name (required):  
  
This value must match the name of the parameter in your report.

Description:

☐ Mandatory  
☐ Read-only  
☒ Visible

**Figure 3-18 Properties of the Text Input Control**



To reuse an input control, add it to the repository independent of any report using **Add Resource > Input Control**. Before using the input control in a report, check that the parameter name in the JRXML matches the name in the Create Input Control page; otherwise, the server can't run the report.

7. Click **Next**.
8. In Locate Datatypes, select **Define a DataType in the next step**, and click **Next**.



Instead of defining a datatype, you can use one in the repository if its type and range are compatible with your input control.

9. In the Set the Datatype Kind and Properties page, enter the properties for the datatype:
  - a. In Type, select the format of the data that the user may enter. Select **Number** from the drop-down.



The number format allows users to enter integers and decimals.

- b. Enter a name – Integer Type
- c. Enter a resource ID – Integer\_Type

The name and resource ID are required, but only visible when defining the input control.

**Add Datatype**

**Set the Datatype Kind and Properties**

First, select the kind of datatype you wish to add, then enter the required property values.

Type: Number

Name (required): Integer Type

Resource ID (required): Integer\_Type

Description:

Minimum value:

Maximum value:

☐ Minimum is Strict

☐ Maximum is Strict

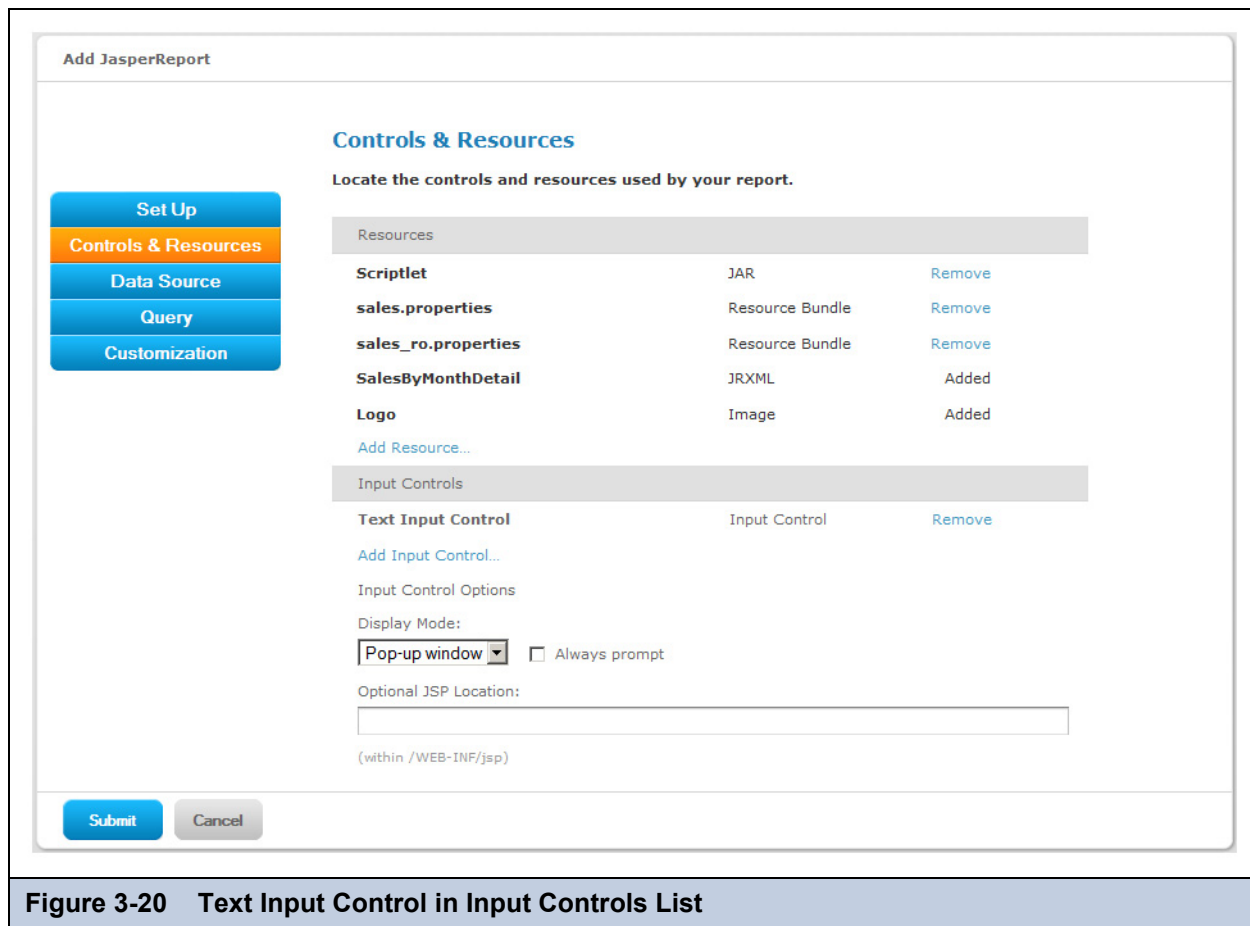
Save Cancel

**Figure 3-19 Integer Datatype Properties**

- d. Skip these properties at the bottom of the Set the Datatype Kind and Properties page:
  - Description – An optional description that only appears within the report wizard: leave blank in this example.
  - Minimum value – The lower bound of the value the user may enter: leave blank in this example.
  - Maximum value – The upper bound of the value the user may enter: leave blank in this example.
  - Minimum is strict – Means the minimum value itself isn't allowed: leave unchecked in this example.
  - Maximum is strict – Means the maximum value itself isn't allowed: leave unchecked in this example.

10. Click **Save**.

The Controls & Resources page now lists the Text Input Control.



**Figure 3-20 Text Input Control in Input Controls List**

### 3.3.2.2 Adding a Simple Check Box Input Control

A check box input control accepts true/false (boolean) input from the user.

**To add a simple check box input control to the complex report example:**

1. Continuing with the previous example, on the Controls & Resources page, click **Add Input Control**.
2. On the Locate Input Control page, click **Define an Input Control in the next step**.
3. Click **Next**.
4. On the Create Input Control page, select the type of input from the **Type** drop-down: **Boolean**.
5. Enter the other properties for the input control.
  - ♦ Prompt Text – Check Box Input Control
  - ♦ Parameter Name – CheckboxInput



Enter the parameter name exactly as shown because the main JRXML file references this name.

- ♦ Description – Leave blank in this example.
  - ♦ Mandatory, Read-only, Visible – Check only visible.
6. Click **Submit**.

The Controls & Resources page appears with the new check box input control.

### 3.3.2.3 Adding a Drop-Down Input Control

The drop-down input control, also called a list input control, gives the user a pre-determined list of choices. As a report designer, you make these decisions about a drop-down input control:

- ♦ To present a single-select or multi-select list to the user
- ♦ To present a single choice as a drop-down list or a set of radio buttons
- ♦ To present a multi-select control as a multi-select list or a set of check boxes

Radio buttons and check boxes usually work well for five or fewer choices. This example shows how to create an input control that presents three choices in a drop-down list. A list of values defines these choices. You can create a new list of values, dedicated to this input control, or you can use a list of values in the repository.

**To add a drop-down input control to the complex report example:**

1. Continuing with the previous example, on the Controls & Resources page, click **Add Input Control**.
2. On the Locate Input Control page, click **Define an Input Control in the next step**.
3. Click **Next**.
4. On the Create Input Control page, select the type of input from the **Type** drop-down: **Single-select List of Values**.
5. Enter the other properties for the input control:
  - ♦ Prompt Text – List Input Control
  - ♦ Parameter Name – ListInput



Enter the parameter name exactly as shown because the main JRXML file references this name.

- ♦ Description – Leave blank in this example.
  - ♦ Mandatory, Read-only, Visible – Check only visible.
6. Click **Next**.
  7. On the Locate List of Values page, select **Define a list of values in the next step**:



Instead of defining a list of values, you can use one in the repository if its values are compatible with the parameter defined in the JRXML report.

8. Click **Next**.
9. On the Add List of Values page, enter a name, resource ID, and optional description for the list of values. These properties aren't visible outside of the input control. Enter these values:
  - ♦ Name – list\_type
  - ♦ Resource ID – list\_type
  - ♦ Description – Leave blank in this example.
10. In the **Name Value** panel, enter names and values to present as choices to the user:
  - ♦ Enter unique names.



The server requires unique names to distinguish which item the user chose.

- ♦ Enter values of the type that match the parameter definition in the JRXML report.

After entering a name and value, click **Add**. If you make a mistake, click **Remove**.

For this example, enter:

- ♦ Name First Item with value 1.
- ♦ Name Second Item with value 2.

- ♦ Name Third Item with value 3.

**Figure 3-21 Definition of the List of Values**

11. Click **Submit**.

The Controls & Resources page appears with the new List Input control.

#### 3.3.2.4 Adding a Date Input Control

In this procedure, you set up a date datatype, and then you define a date input control using this datatype.

**To add a date input control to the complex report example:**

1. On the Controls & Resources page, click **Add Input Control**.
2. On the Locate Input Control page, select **Define an Input Control in the next step**.
3. Click **Next**.
4. On the Create Input Control page, select the type of input from the **Type** drop-down: **Single-Value**
5. Enter the other properties for the input control:
  - ♦ Prompt Text – Date Input Control
  - ♦ Parameter Name – DateInput



Enter the parameter name exactly as shown because the main JRXML file references this name.

- ♦ Description – Leave blank in this example.
  - ♦ Mandatory, Read-only, Visible – Check only the visible setting.
6. Click **Next**.
  7. On the Locate Datatypes page, select **Define a Datatype in the next step**.
  8. Click **Next**.
  9. In the Set the Datatype Kind and Properties page, set the following properties:
    - a. In Type, select **Date**.
    - b. In Name, enter: Date Datatype
    - c. In Resource ID, enter: DateDatatype
    - d. Skip properties at the bottom of the page. For information about these properties, see [step d in 3.3.2.1, "Adding a Text Input Control," on page 45](#).
  10. Click **Save**.



The Controls & Resources page appears with the new Date Input Control.

### 3.3.2.5 Adding a Query-Based Input Control

A query-based input control presents a dynamically-created list of choices to the user. The server performs a query whose results are used to create the list of choices. You must perform the following tasks:

- Configure the query.
- Designate how to display the results in the input control.
- Specify the value to pass as the corresponding parameter.

**To add a query-based input control to the complex report example:**

1. On the Controls & Resources page, click **Add Input Control**.
2. On the Locate Input Control page, select **Define an Input Control in the next step**.
3. Click **Next**.
4. On the Create Input Control page, select the type of input from the **Type** drop-down: **Single-select Query**.
5. Enter the naming properties for the input control:
  - Prompt Text – Query Input Control
  - Parameter Name – QueryInput



Enter the parameter name exactly as shown because the main JRXML file references this name.

- Description – Leave blank in this example.
  - Mandatory, Read-only, Visible – Use the default settings in this example.
6. Click **Next**.  
The Locate Query page appears. Options are:
    - To locate a reusable query in the repository
    - To define a new query dedicated to this input control
  7. For this example, select **Define a Query in the next step**, and click **Next**.
  8. On the Name the Query page, enter naming properties for the new query. For this example, enter `testQuery` in both the Name and Resource ID fields.

**Add Query**

**Name the Query**

Enter the required property values.

- **Name the Query**
- Link a Data Source
- Define the Query

Name (required):

Resource ID (required):

Description:

**Figure 3-22 Properties of the Query for the Query Input Control**

9. Click **Next**.  
The Link a Data Source to the Report page appears. Options are:
  - To use the same data source for the input control as you use for the report

- ♦ To define a new data source, dedicated to this input control
  - ♦ To select a reusable data source from the repository
10. For this example, select **Do not link a data source** to use the same data source for the input control as you use for the report. You will select the data source for the report in 3.3.3, “[Selecting a Data Source and Running the Complex Report,](#)” on page 52.

The screenshot shows the 'Add JasperReport' dialog box with the 'Link a Data Source to the Report' step selected. The left sidebar shows 'Name the Query', 'Link a Data Source' (selected), and 'Define the Query'. The main area has the title 'Link a Data Source to the Report' and the instruction 'Optionally, link a data source to the report. You can choose a data source from the repository or create a new one.' There are three radio buttons: 'Do not link a data source.' (selected), 'Click here to create a new data source.', and 'Select data source from repository.' Below the radio buttons is a text input field and a 'Browse...' button. At the bottom are 'Previous', 'Next', and 'Cancel' buttons.

**Figure 3-23 Data Source Link for the Query Input Control**

11. Click **Next**.
12. On the Define the Query page, select **SQL** from the Query Language drop-down.
13. Enter the following Query String to retrieve the labels and values for this input control:
- ```
SELECT user_name, first_name, last_name FROM users
```

The screenshot shows the 'Add Query' dialog box with the 'Define the Query' step selected. The left sidebar shows 'Name the Query', 'Link a Data Source', and 'Define the Query' (selected). The main area has the title 'Define the Query' and the instruction 'Select a language and enter the query.' There is a 'Query Language:' dropdown menu with 'SQL' selected. Below it is a 'Query String:' text input field containing the text 'SELECT user\_name, first\_name, last\_name FROM users'. At the bottom are 'Previous', 'Save', and 'Cancel' buttons.

**Figure 3-24 Query String Definition**

14. Click **Save**.
15. For each row of results returned by the query, the server presents a single value, such as Sarah Smith, in the input control widget (drop-down, radio buttons, mutli-choice, check boxes). On the Query Information page, name the database columns to construct this value. The column names must match those in the `SELECT` clause of the query string exactly.
- In the Value Column, enter the `user_name`.
  - In the Visible Column, enter `first_name`.
  - Click **Add**.
  - In the Visible Column, enter `last_name`.

- e. Click **Add**.

For each visible column that you want to display as a choice, enter the name, then click **Add**. If you make a mistake, click **Remove**. **Figure 3-25** shows the Query Information page.

**Add Input Control**

**Query Information**

Provide parameters for the value column and other columns you wish to appear in the selector.

Value Column

user\_name (required)

Visible Columns

|            |        |
|------------|--------|
| first_name | Remove |
| last_name  | Remove |
| last_name  | Add    |

Previous Submit Cancel

**Figure 3-25** Column Parameters for the Query Input Control

16. Click **Submit**. The Controls & Resources page displays all the resources, including the new input controls.

**Add JasperReport**

**Controls & Resources**

Locate the controls and resources used by your report.

Resources

|                     |                 |        |
|---------------------|-----------------|--------|
| Scriptlet           | JAR             | Remove |
| sales.properties    | Resource Bundle | Remove |
| sales_ro.properties | Resource Bundle | Remove |
| SalesByMonthDetail  | JRXML           | Added  |
| Logo                | Image           | Added  |

[Add Resource...](#)

Input Controls

|                         |               |        |
|-------------------------|---------------|--------|
| Text Input Control      | Input Control | Remove |
| Check Box Input Control | Input Control | Remove |
| List Input Control      | Input Control | Remove |
| Date Input Control      | Input Control | Remove |
| Query Input Control     | Input Control | Remove |

[Add Input Control...](#)

Input Control Options

Display Mode: Pop-up window ☐ Always prompt

Optional JSP Location:

(within /WEB-INF/jsp)

Submit Cancel

**Figure 3-26** Input Controls and Resources

Next, set the input control options.

### 3.3.2.6 Setting the Input Control Options

In this procedure, you set the display mode in **Input Control Options** at the bottom of the Controls & Resources page. These options only appear after you add an input control to the report. [Figure 3-26](#) shows these options.

**To configure the appearance of the input controls for the complex report example:**

1. Select **Pop-up window**.



You can also select **Separate page** to display the input controls in a separate browser window, **Top of page** to display them above the report, or **In page** to display them on the side of the report.

2. Check **Always prompt** when you want the server to display the Input Controls dialog to prompt the user when the report runs.



The definition of input controls in this example specified Visible and not Mandatory. When input controls aren't mandatory and Always prompt isn't checked on the Controls & Resources page, the user must click the Options button in the Report Viewer to change input controls; otherwise, the report runs with default input controls.

3. Leave **Optional JSP Location** blank for this example.

You can use the Optional JSP Location option to specify the path to a JSP file that affects the appearance of the input controls.

Select the data source to finish the complex report example.

### 3.3.3 Selecting a Data Source and Running the Complex Report

You select a data source to retrieve data for the report and the query input control; otherwise, the report and the list of users in the query input control will be blank.


**To select a data source and run the complex report:**

1. On the Controls & Resources page of the JasperReport wizard, select Data Source.
2. On the Locate Data Source page, choose **Select data source from repository**.
3. Click **Browse**, and choose **Root > Data Sources > JServerJNDI Data Source**. Click the **Select** button.
4. On Link a Data Source to the Report, click **Submit**.
5. On the Locate Query Page, click **Submit** again to save the complex report.

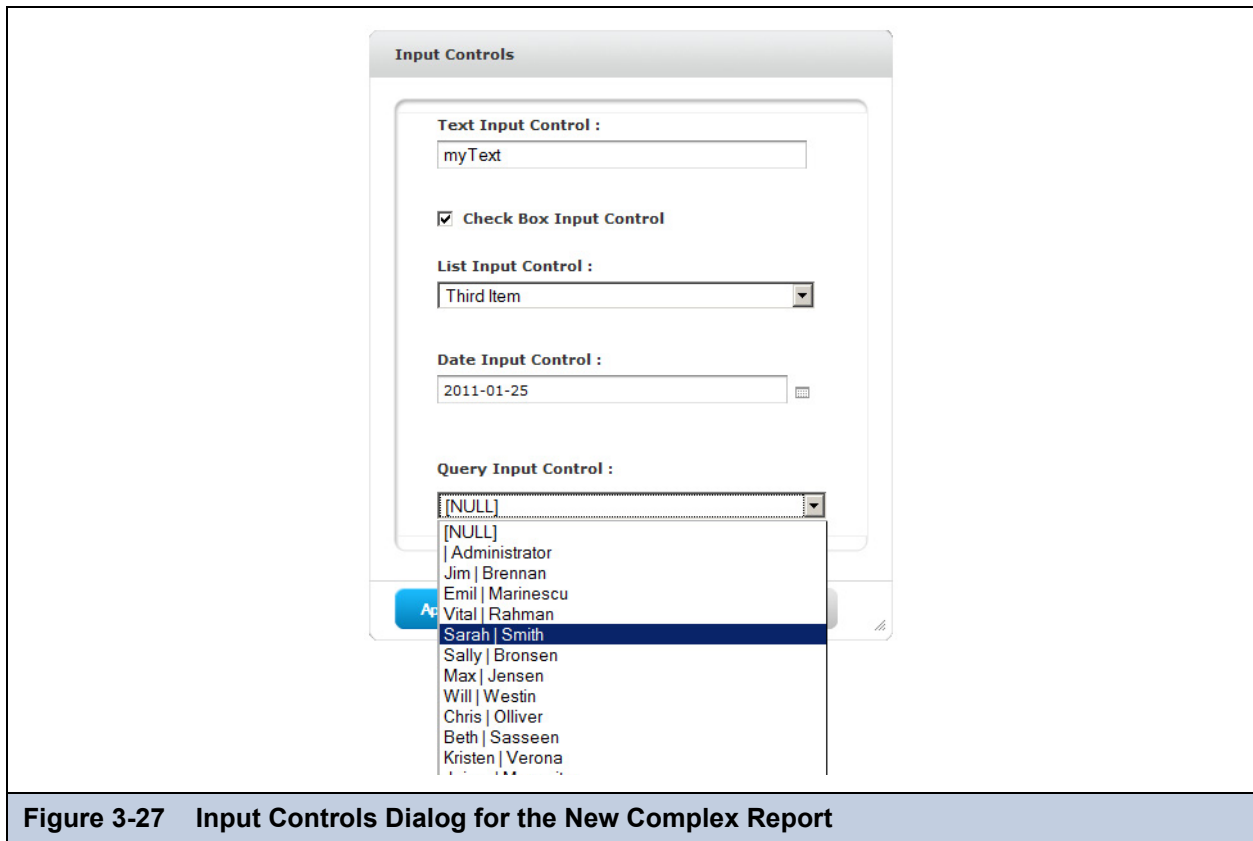
Skip the Query and Customization pages of the JasperReports wizard to use the default settings on those pages.

The server validates the report and a message appears indicating that the report was added to the repository. The New Complex Report appears in the repository.

6. In the Repository, click the name New Complex Report to run and view the report.  
Input controls appear.
7. Enter these input values, as shown in [Figure 3-27](#):

- a. Text Input Control: myText
- b. Check Box Input Control: Check the checkbox.
- c. List Input Control: Select **Third Item**.
- d. Date Input Control: Click , select December 31, 2010.

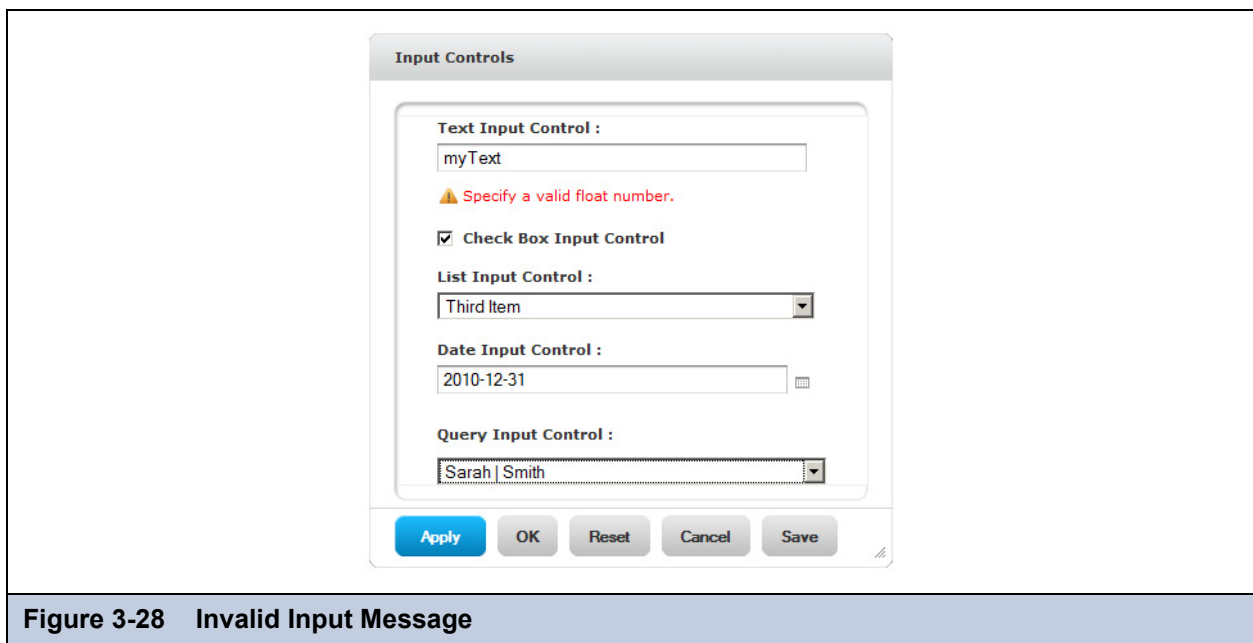
- e. Query Input Control: Select **Sarah Smith** from the drop-down.



**Figure 3-27 Input Controls Dialog for the New Complex Report**

8. Click **OK** or **Apply** to run the report with the selected input, including the incorrect non-numerical input for the Text Input Control.

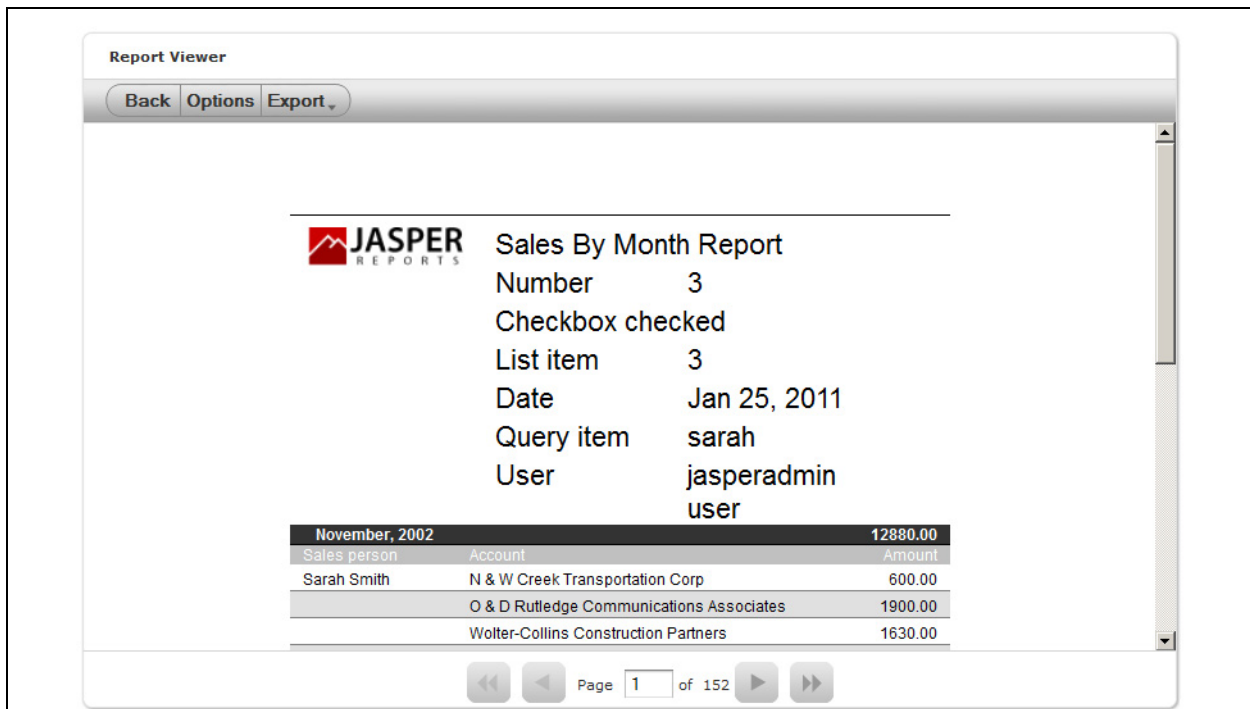
The server enforces the proper format defined for each input control. You defined the Text Input Control as a numeric type, so it accepts only valid numbers, as indicated by the message to specify a valid float number, as shown in [Figure 3-28](#).



**Figure 3-28 Invalid Input Message**

9. In Text Input Control, enter 3 and click **OK** or **Apply**.

The sample report output includes a header that displays the value of each parameter received from the input controls. Values and labels for the input controls appear in the language specified by the active resource bundle, in this case English.




**Figure 3-29 Output Controlled by Input**

In the Report Viewer, you can open the Input Controls dialog at any time by clicking the **Options** button. Click **OK** to run the report using the chosen values and close the Input Controls dialog; click **Apply** to run the report using the chosen values, but keep the Input Controls open for choosing other values and re-running the report.



If you get an error when you run the report, open it for editing as described in [3.5, “Editing JRXML Report Units,” on page 55](#). Review your settings. If you can’t find the problem, edit the SalesByMonth sample report (in the repository at /reports/samples) and compare its settings to your report.

To see the message written by the scriptlet JAR on the last page of the report, click  in the Report Viewer.

### 3.4 Adding Cascading Input Controls to a Report

JRXML-based reports can include input controls that have dynamic values. The values depend on a user's selection in other input controls. For example, a report has input controls for country, state, and city. The options in the State input control depend on the value selected in the Country input control. When the user selects a state, the list of values includes only those in the selected state. These cascading input controls use queries to determine the values to display in each input control field.

To use input controls as parameters for a query that populates another input control, you use a special syntax to reference a parameter name in the input control's query. The syntax is identical to the `$P{parameterName}` syntax used in queries for JasperReports. For more information about defining the input control's query, see section [3.3.2.5, “Adding a Query-Based Input Control,” on page 49](#).

For example, a report returns data identified by country and city. It includes input controls called COUNTRY and CITY. This query used by the CITY input control retrieves city names from a table called `accounts` based on the value selected by the COUNTRY input control:

```
select city from accounts where country = $P{COUNTRY}
```

When the user selects a country from the COUNTRY input control, the value selected is used by the query of the CITY input control; the CITY input control is refreshed to show the result. Note that there are other ways to use a parameter in a query.

For an example of a report that has cascading input controls, see section 2.2.2, “Cascading Input Controls,” on page 18.

### 3.5 Editing JRXML Report Units

After you add a report unit to the repository, you can edit any of its elements, including file resources and input controls. This example modifies the display text of the ambiguous Text Input Control.

**To edit the complex report example:**

1. Log into the server as an administrator and select **View > Repository**.



If you log in as a user, you can edit a report that you created. This example requires an administrator login because an administrator created the complex report.

2. Search or browse the repository to locate the report. In this example, go to **Root > Reports > Samples**.
3. Right-click the New Complex Report and select **Edit** from the context menu.  
The JasperReport wizard opens the report.
4. Navigate to the page of the wizard for making the change; in this example, click **Controls & Resources**.
5. Make changes to an input control prompt and the display mode of the input controls, for example:
  - a. Click the name of the **TextInput** control.  
The Locate Input Control page shows that this input control is locally defined.
  - b. Click **Next**.  
The Create Input Control page appears.
  - c. Change the contents of the Prompt Text field to `Enter a number`.  
The Locate Datatypes page appears. You can select a different datatype from the repository. For this example, accept the existing datatype setting.
  - d. In Locate Datatypes, click **Next**.
  - e. In Set the Datatype Kind and Properties, click **Save** to accept the datatype property settings.
6. On the Controls & Resources page:
  - a. Change the Display Mode to **In Page**.
  - b. Clear the **Always prompt** check box.
  - c. Click **Submit**.
7. Run the New Complex Report again.

Instead of appearing in a pop-up before the report, the input controls appear in the Options panel of the report.

**Figure 3-30** shows the new prompt, **Enter a number**, for the text input control. Because none of the input controls in this example are required, the report can run with blank input controls. Enter values and click **Apply** to modify the report output according to your input.

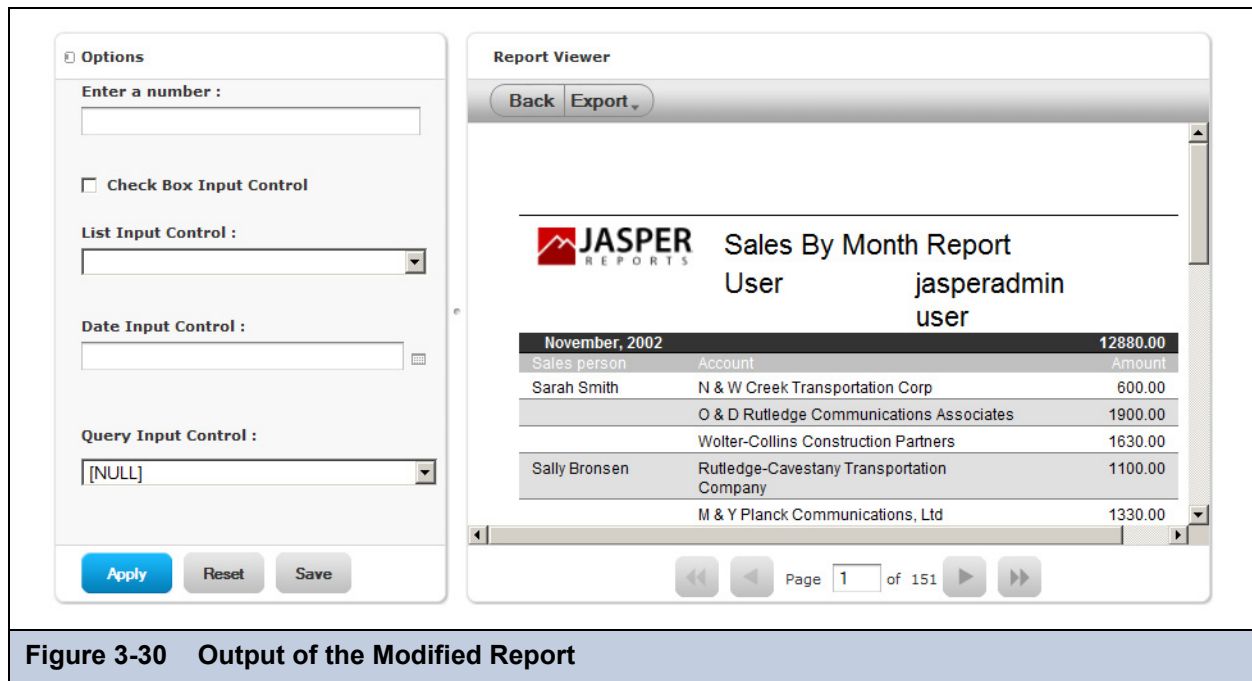


Figure 3-30 Output of the Modified Report

## 3.6 Localizing Reports

You can adapt reports to global audiences by localizing input controls and field names:

- Input controls – The server supports multi-lingual prompts and static lists of values in reports.
- Field names – The server supports multi-lingual field names in reports.

A `$R` expression that you write in the report design triggers linguistic changes in the report output for different locales. Each `$R` expression refers to a name-value pair (your translations) in a resource bundle. A resource bundle file is a text file that has a `.properties` extension. You can create a resource bundle in iReport or a text editor. You set the base name of the resource bundle in the header of the JRXML file:

```
<jasperReport name="StoreSales" pageWidth="595" pageHeight="842" columnWidth="515"
leftMargin="40" rightMargin="40" topMargin="50" bottomMargin="50"
resourceBundle="simpleTable">
```

For example, `simpleTable` is the base name of the resource bundle file for this report. If you prefer using a graphical user interface to coding in XML, use iReport Designer to set the base name of the resource bundle. You name of the default English resource bundle `simpleTable.properties`, and the French resource bundle, `simpleTable_fr.properties`. For more information about these naming conventions, see *JasperReports Server Community Project Localization Guide*.

### 3.6.1 Running a Localized Report


In this procedure, you run the Romanian version of the complex report that you added in section 3.3.1, “[Uploading Undetected File Resources](#),” on page 40.

**To run the Romanian version of the complex report:**

1. Choose the Romanian locale on the login page of the server, and login as an administrator.
2. Click **View > Repository**, and navigate to **Root > Reports > Samples**.
3. Click the name of the complex report, New Complex Report.

The Input Controls dialog appears.

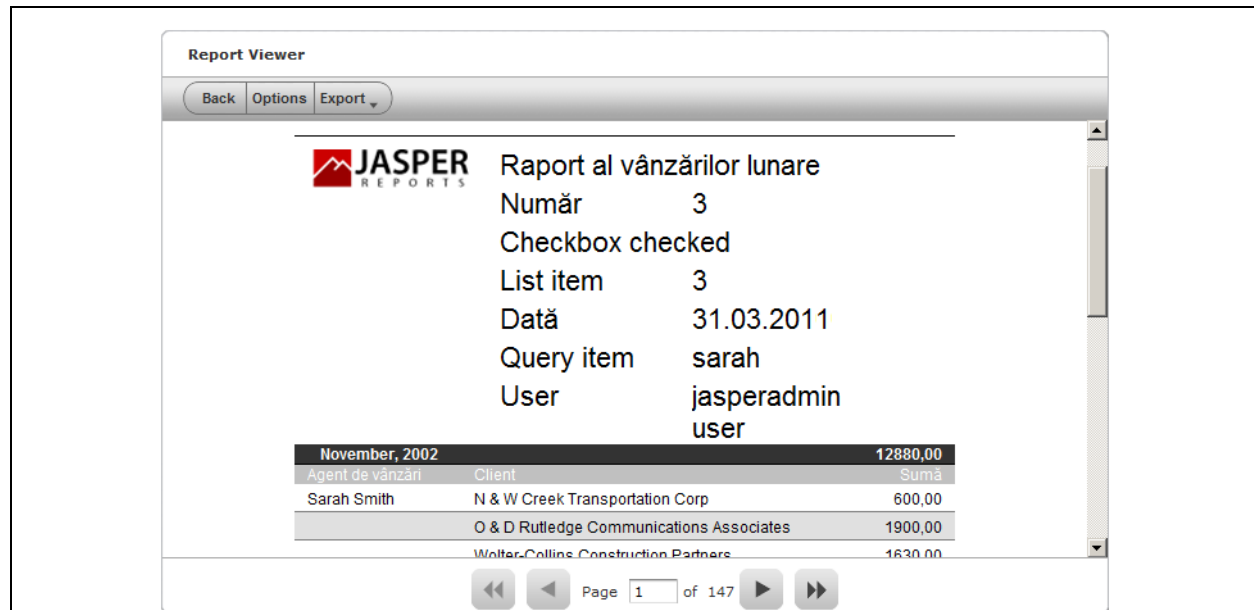


4. Enter input control values:
  - a. Text Input Control: 3
  - b. Check Box Input Control: Check the check box.
  - c. List Input Control: Select **Third Item**.
  - d. Date Input Control: Click , select March 31, 2011.
  - e. Query Input Control: Select **Sarah Smith** from the drop-down.
  - f. Click **OK**.

The fields in the title band and column names (sales person, sales account, and sales amount), shown in [Figure 3-31](#), appear in the language set by the Romanian resource bundle: sales\_ro.properties.

```
title=Raport al v\u00E2nz\u0103rilor lunare
sales.person=Agent de v\u00E2nz\u0103ri
sales.account=Client
sales.amount=Sum\u0103
param.number=Num\u0103r
param.date=Dat\u0103
```

The currency and dates in the report output header map to Romanian locale settings.



**Figure 3-31 A Report Localized for the Romanian Locale**

By default, the web interface elements appear in US English when you chose an unsupported locale, such as the Romanian locale. If you choose a supported language, the web interface elements appear in that language. Supported languages are Chinese (Simplified), French, German, Japanese, and Spanish. You can customize the server to support additional languages; you can translate the web interface into a different language, server property names, and messages in another language. For some locales, you may also need to change the default locale and time zone. For more information about localizing the server, see the *JasperReports Server Community Project Localization Guide*.

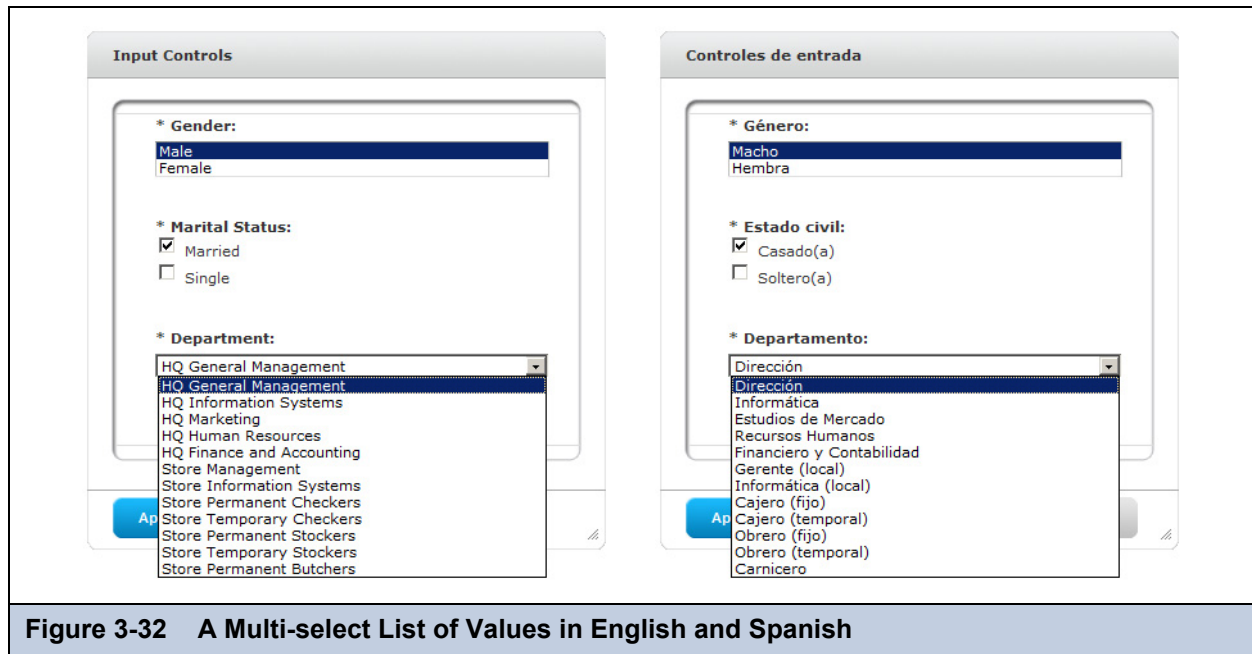
### 3.6.2 Adding a Multi-Lingual Prompt and Static List to an Input Control

The server provides linguistic support for the prompt and list of values that an input control can present to users. The input controls that can present a list of values are:

- ♦ Single-select list of values
- ♦ Single-select list of values (radio)

- Multi-select list of values
- Multi-select list of values (check box)

**Figure 3-32** shows an example of the multi-lingual Department Report in the Reports/Samples folder (if you installed sample reports during installation). The prompts and the multi-select list of values appear in English or Spanish, depending on the locale you select on the login page.



**Figure 3-32 A Multi-select List of Values in English and Spanish**

This next example shows how to localize an input control to make the prompts and static list of values multi-lingual. These tasks are explained step-by-step:

- Edit the list input control to localize prompts and a static list of values.
- Create a new resource bundle for the French translations.
- Upload the French resource bundle.
- Create a new resource bundle for the English translations.
- Replace the existing English resource bundle with the new one.
- Remove unwanted parameters from the sample report design.
- Run the report that uses the localized input control.

You edit a sample report to add `$R` expressions to the list input control definition.

#### To edit the list input control to localize prompts and a static list of values:

1. Log into the server as administrator and add a folder to the repository named **Temp**.
2. Locate the Sales By Month report in the **Reports > Samples** folder of the repository. Select it and click **Copy**.
3. Navigate to the **Temp** folder and choose **Paste**.  
The Sales By Month Report appears in the Temp folder.
4. Select the Sales By Month Report and click **Edit**.  
The JasperReport wizard appears.
5. Click **Controls & Resources**.  
The report has five input controls, including a List Input Control, two resource bundles, and other resources, as shown in [Figure 3-26 on page 51](#).
6. Click the **List Input Control** to edit it.
7. In Locate Input Control, click **Next**.
8. On the Create Input Control page, change the prompt text from List Input Control to `$R{department_multi}`.

**Figure 3-33** shows the `$R` expression in the **Prompt Text** field.

**Add Input Control**

**Create Input Control**

First, select the kind of input control you wish to add, then enter the required property values.

Type: Single-select List of Values

Prompt Text (required):  
  
 The label is displayed to users next to the input control.

Parameter Name (read-only):  
  
 This value must match the name of the parameter in your report.

Description:

Previous Next Cancel

**Figure 3-33 A \$R Expression in the Prompt Text Field**

9. Click **Next**.
10. On the Locate List of Values page, click **Next**.  
 The Edit List of Values page appears.
11. On the Edit List of Values page, click **Remove** to remove each existing name-value pair.
12. Add these name-value pairs:

| Name                           | Value |
|--------------------------------|-------|
| <code>\$R{department_1}</code> | 1     |
| <code>\$R{department_2}</code> | 2     |
| <code>\$R{department_3}</code> | 3     |

**Figure 3-34** shows the edited list of values containing the \$R expressions in the name fields.

**Edit List of Values**

Identify the list, then create the name-value pairs.

Name (required):

Resource ID (read-only):

Description:

| Name                            | Value                |                        |
|---------------------------------|----------------------|------------------------|
| <code>\$R {department_1}</code> | 1                    | <a href="#">Remove</a> |
| <code>\$R {department_2}</code> | 2                    | <a href="#">Remove</a> |
| <code>\$R {department_3}</code> | 3                    | <a href="#">Remove</a> |
| <input type="text"/>            | <input type="text"/> | <a href="#">Add</a>    |

Submit Cancel

**Figure 3-34 The Revised List of Values Containing \$R Expressions**

13. Click **Submit**.

The Controls & Resources page shows the change to the input control prompt. The expression `$R{department_multi}` replaces List Input Control in the list of input controls.

14. Click **Submit** again to save your work.**To create a new resource bundle for the French translations:**

1. In a text editor, create a new file for French translations, and enter these name-value pairs for the input control:

```
department_multi=Choisir le service
department_1 = Travail # 1
department_2 = Travail # 2
department_3 = Travail # 3
```

2. At the bottom of the file, enter these field and property translations to localize the report:

```
title=Rapport de ventes mensuelles
sales.person=Commercial
sales.account=Compte
sales.amount=Montant
param.list=Liste Détaillée
```



Use the Unicode escape sequence if you can't enter é: `\u00e9`

3. Save the file as `sales_fr.properties` to your hard drive.

**To upload the French resource bundle:**

1. Assuming you're still logged into the server, select the Sales By Month report, and click **Edit**.
2. On the Controls & Resources page, click **Add Resource**.
3. Select **Upload a Local File**, click **Browse**, and locate the `sales_fr.properties` file on your hard drive. Select the file, and click **Open**.
4. In Locate File Resource, click **Next**.  
On the Add a Report Resource page, `sales_fr.properties` appears as the Selected Resource, indicating that the server automatically detected it as a resource bundle.
5. On the Add a Report Resource page, enter this information:
  - ♦ Name – `sales_fr.properties`
  - ♦ Resource ID – `sales_fr.properties`



Enter the Resource ID field exactly as shown.

6. Click **Next**.

The Controls & Resources page appears.

**To create a new resource bundle for the English translations:**

1. In a text editor, create a new file for English translations, and enter these input control name-value pairs in the file:

```
department_multi=Select Department
department_1=Work #1
department_2=Work #2
department_3=Work #3
```

- At the bottom of the file, enter these field and property translations to localize the report:

```
title=Sales By Month Report
sales.person=Sales person
sales.account=Account
sales.amount=Amount
param.list=List item
```

- Save the file as `sales.properties` to your hard drive.

**To replace the old English resource bundle in the repository with the new one:**

- Assuming you're still logged into the server and editing the Sales By Month report, remove the old `sales.properties` resource bundle:
  - On the Controls & Resources page, select `sales.properties`.
  - Click **Remove**.
- Click **Add Resource**.
- On the Locate File Resource page, select **Upload a Local File**, click **Browse**, and locate the new `sales.properties` file that you created in the previous procedure. Click **Open**.
- Click **Next**.  
On the Add a Report Resource page, `sales.properties` appears as the Selected Resource, indicating that the server automatically detected it as a resource bundle.
- On the Add a Report Resource page, enter this information:
  - Name – `sales.properties`
  - Resource ID – `sales.properties`



Enter the Resource ID field exactly as shown.

- Click **Next**.
- On Controls & Resources, click **Submit**.  
At this point, the report doesn't run because the `sales.properties` file doesn't contain name-value pairs for all parameters that the JRXML defines. In the next procedure, you remove these parameters from the report.

**To remove unwanted parameters from the sample report design:**

In this procedure, you connect to the server from iReport using the JasperReports Server Plug-in, then you remove elements in the title band that lists parameter names and settings in the report output header.


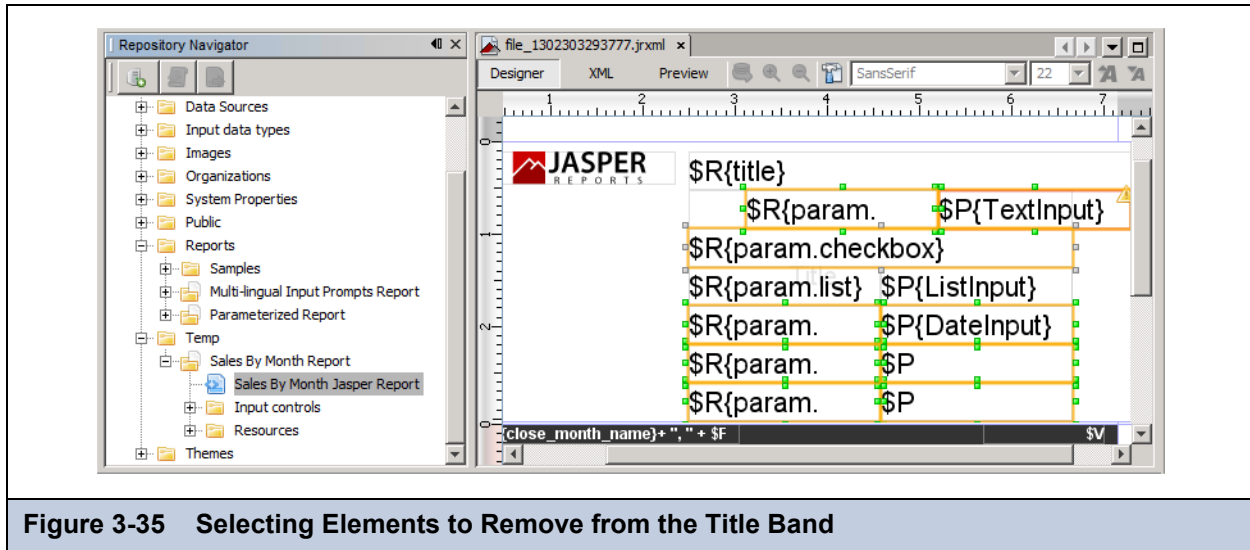
- Connect to the server from iReport, as described in section 6.2, “Connecting to the Server from iReport,” on page 94.
- In the Repository Navigator, expand the Temp and Sales By Month Report folders, then double-click  Sales By Month Jasper Report.  
The JRXML report appears in the Designer tab.
- Ctrl-click to select all elements in the title band except `$R{title}`, `$R{param.list}`, and `$P{ListInput}`.

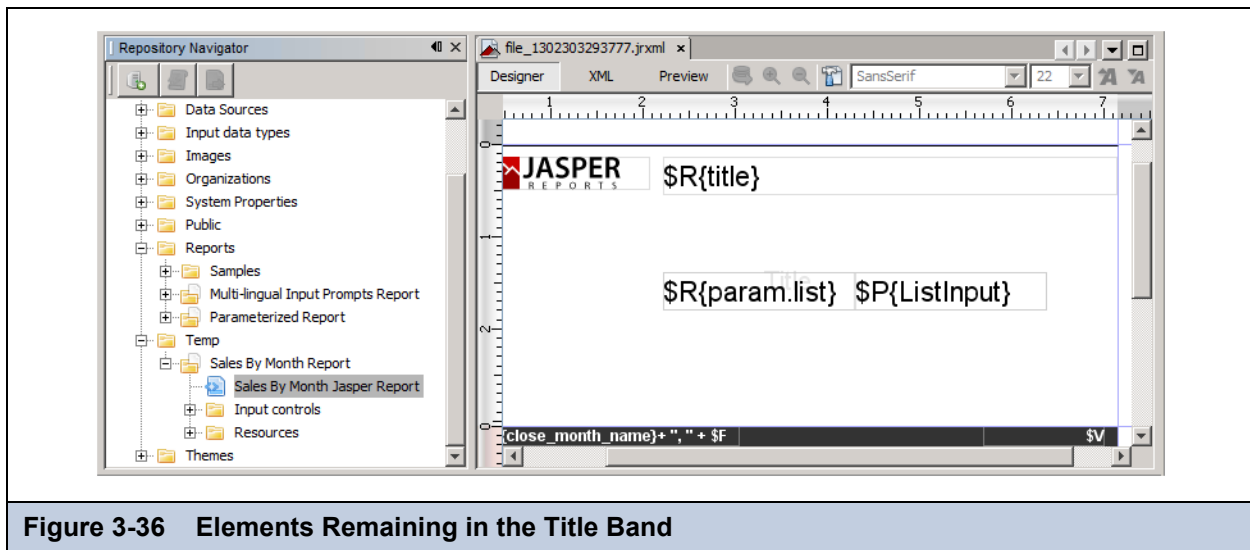
Figure 3-35 shows the selected elements in the title band.



**Figure 3-35 Selecting Elements to Remove from the Title Band**

4. Right-click an empty area in the Designer tab, and choose Delete. If you have empty elements after the last step, delete them.

Figure 3-36 shows the remaining elements.

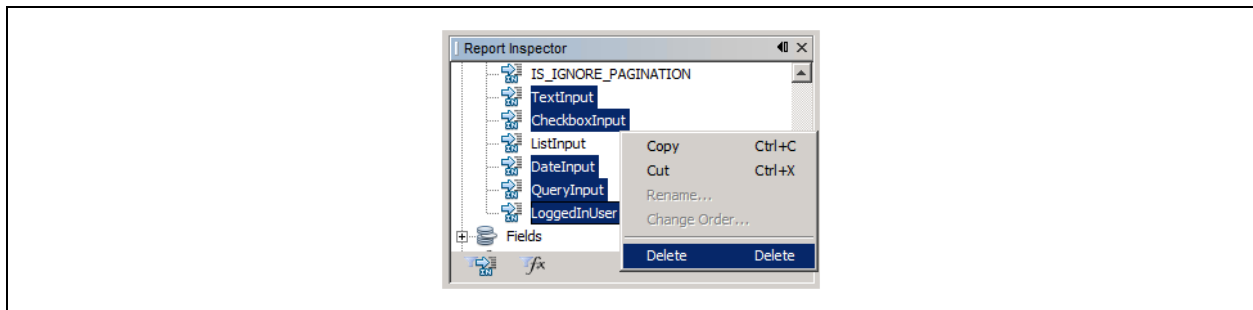


**Figure 3-36 Elements Remaining in the Title Band**

There's too much white space in the title band.

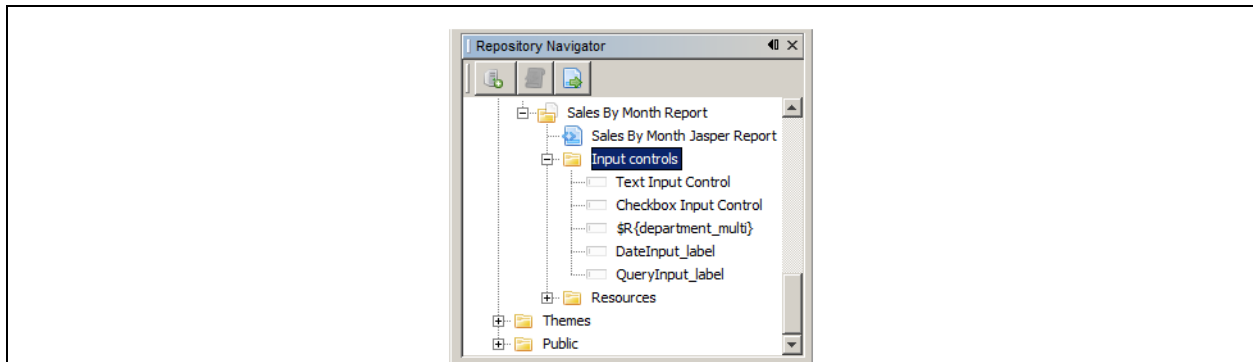
5. To remove excess white space:
  - a. Drag a selection box around the `$R{param.list}` and `$P{ListInput}` elements.
  - b. Press the **Arrow** key on the keyboard to move the elements upward.
  - c. Drag the bottom margin of the title band upward to reduce its height.
6. Remove the actual parameters from the report:
  - a. Choose **Window > Report Inspector**.
  - b. In the Report Inspector, expand Parameters, scroll to the bottom of the parameter list, and Ctrl-click to select these parameters:
    - ♦ `TextInput`
    - ♦ `CheckboxInput`
    - ♦ `DateInput`

- ♦ QueryInput
  - ♦ LoggedInUser
- c. Right-click the selected parameters and select **Delete**.



**Figure 3-37 Deleting Unwanted Parameters From the JRXML**

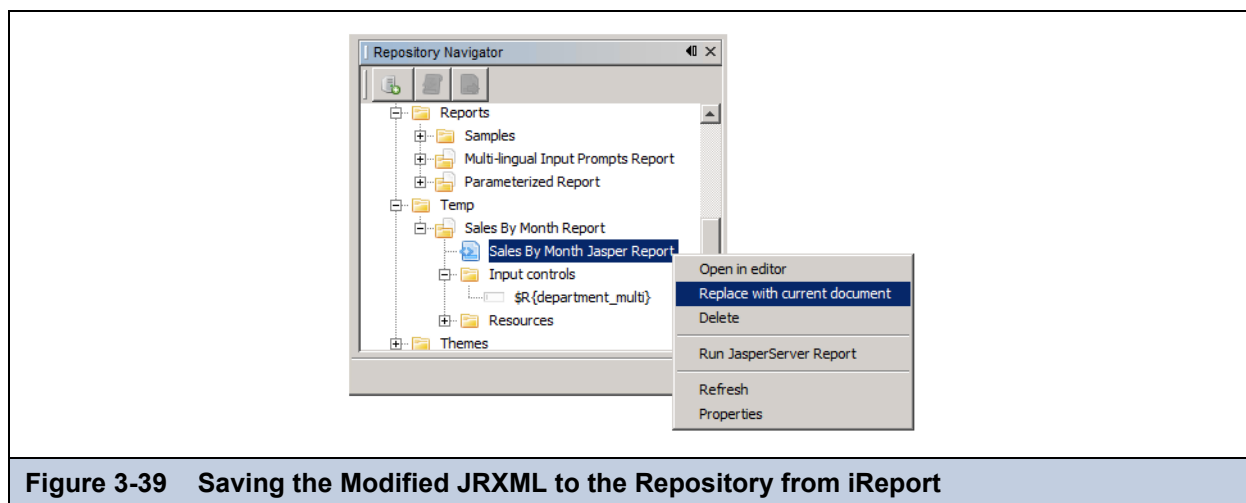
7. Click **Yes** to confirm that you want to delete these 5 items.
8. From iReport, remove the input control resources from the report unit in the repository.
- a. In the Repository Navigator, expand Input Controls in the report unit Sales By Month Report.



**Figure 3-38 Expanding the Input Controls Folder**

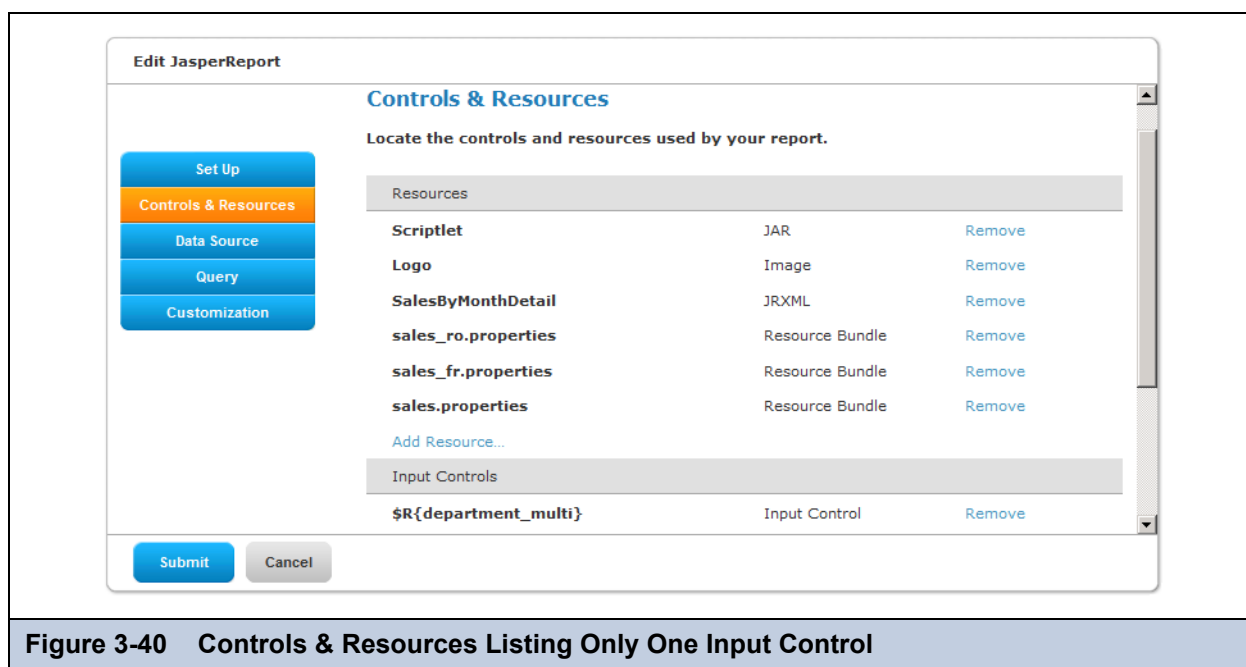
- b. Ctrl-click to select these input control resources:
- ♦ Text Input Control
  - ♦ Checkbox Input Control
  - ♦ DateInput\_label
  - ♦ QueryInput\_label
- c. Right-click the selected resources and click **Delete**
- Answer **Yes** to each prompt to confirm the deletion of the resource.

- Right-click the Sales By Month Jasper Report JRXML, and choose **Replace with Current Document**.



**Figure 3-39 Saving the Modified JRXML to the Repository from iReport**

- Click **Yes** when asked if you want to save the file before sending its content to the server.
  - Click **View > Repository**, navigate to the Temp folder, and select the Sales By Month Report. Click **Edit**.
- Figure 3-40** shows the JasperReport wizard on the server, which lists one remaining input control in the report.



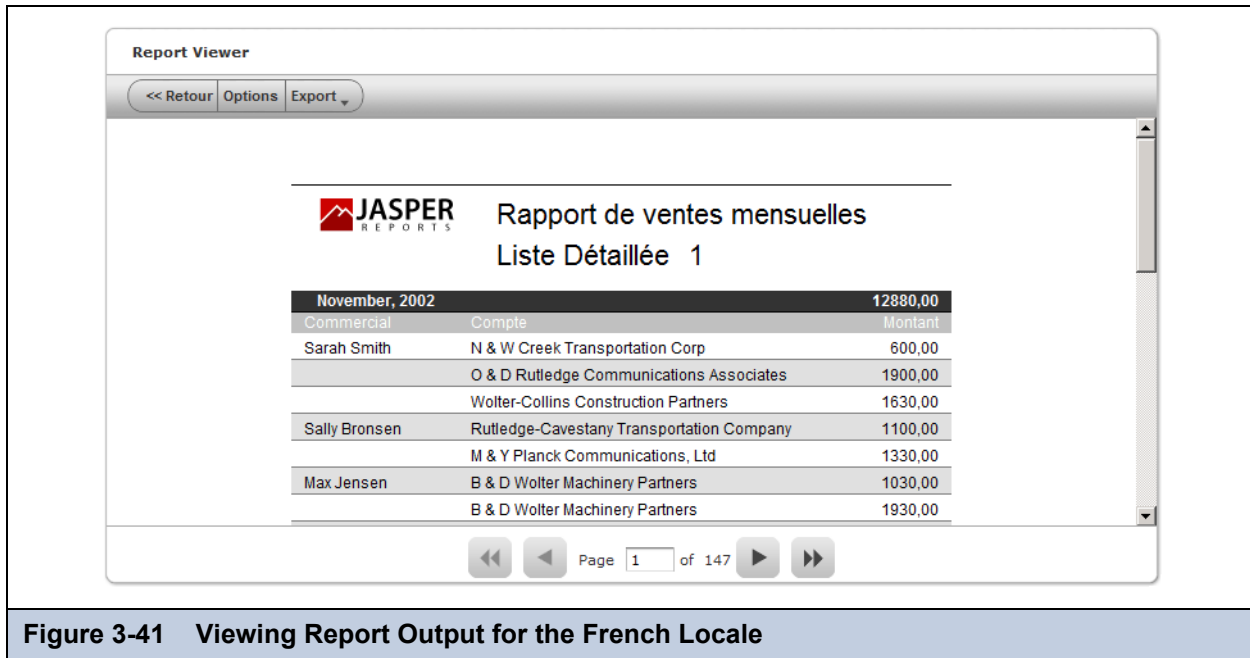
**Figure 3-40 Controls & Resources Listing Only One Input Control**

**To run the report that uses the localized input controls:**

- Click LogOut. On the server login page, click **Show locale & time zone**, select the **en\_US-English (United States)** locale, and log into the server as an administrator.
- Navigate to the Temp folder of the repository, and click the name of the Sales By Month Report to run the report. The Report Viewer presents the report using English field and property names in the report.
- Click **Options**, and select the drop-down. The prompt and list of values appear in English, as shown in **Figure 3-42 on page 65**.
- Logout, on the server login page, click **Show locale & time zone**, select the **fr\_French** locale, and log into the server as an administrator.
- Navigate to the **Temp** folder of the repository, and click the name of the Sales By Month Report to run the report.



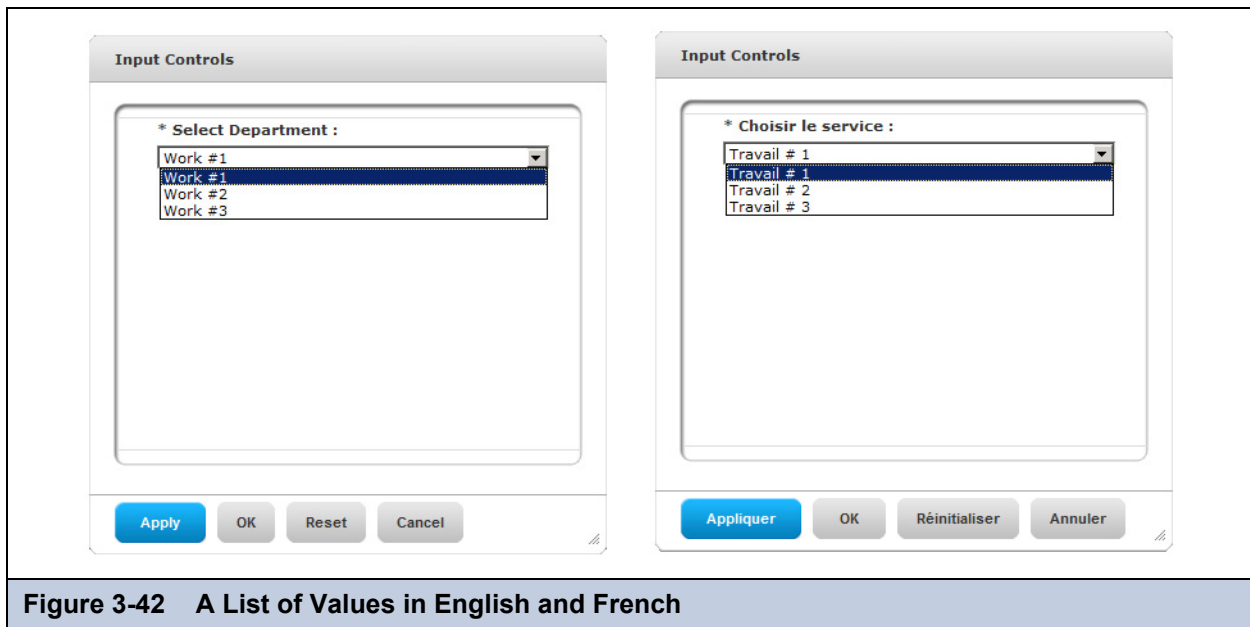
**Figure 3-41** shows the report output: the French report title, field, and property names defined in the resource bundle. The server formats the currency using a comma instead of a decimal point to separate Euro cents from dollars. Translation packs and resource bundles installed with the server for the French locale are used to format currency and dates and to translate web interface components.



**Figure 3-41 Viewing Report Output for the French Locale**

- Click **Options**, and select the drop-down. The input control prompt and static list of values appears in French.

**Figure 3-42** shows the input control prompt and static list of values in English and in French.



**Figure 3-42 A List of Values in English and French**

### 3.6.3 Reusing Resource Bundle

The location of a resource bundle determines its capability for being reused and conditions under which the server uses it. To resolve a \$R expression in a report, the server scans resource bundles at two levels in the order described in this table.

| Order | Level                 | Description                                                                                                                                                                                                                                                                                                                                |
|-------|-----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1     | Report/<br>Repository | A report-level bundle declares the resource bundle base name in the header of its JRXML file. The Romanian resource bundle is a report-level bundle. A repository-level bundle is independent of any specific report and can be linked to multiple reports, as described in section 7.5, “ <a href="#">Locale Bundles</a> ,” on page 200). |
| 2     | Server                | A server-wide bundle typically contains server messages and labels of user interface components, as described in the <i>JasperReports Server Administrator Guide</i> . This bundle typically resides in the WEB-INF/bundles directory of the server.                                                                                       |

First, the server searches the report/repository level and stops scanning resource bundles if it finds a resolution to the \$R expression. If the server doesn’t find a resolution, it scans the server level for the resource ID of the field and uses this ID.

### 3.6.4 Using Default Fonts in JasperReports Server

By default, the server uses three fonts you can use in reports:

- DejaVu Sans
- DejaVu Serif
- DejaVu Sans Mono

Using the DejaVu fonts shipped with the server ensures availability of fonts in all environments; the PDF is pixel-perfect every time.

The DejaVu fonts replace the Java logical fonts used in previous versions of the server:

- SansSerif
- Serif
- Monospaced



SansSerif, Serif, Monospaced can still be used, but are deprecated because these Java logical fonts map to different TTF files in different environments, and run the risk of text being cut when exported to PDF due to font metric mismatches. Also, these Java logical fonts aren't recognized by some browsers, resulting in font substitution. For example, Firefox in a Windows environment renders the SansSerif logical font as Serif.

When using the DejaVu fonts coming from JasperReports font extensions, you don’t need to set any other font attributes (such as the pdfXXX attributes) in the JRXML or specify font mapping. The font extension file sets font attributes and mapping.

For more information about DejaVu fonts, refer to its SourceForge project at: [http://dejavu-fonts.org/wiki/index.php?title=Main\\_Page](http://dejavu-fonts.org/wiki/index.php?title=Main_Page).



When you upload a TrueType font to the repository, the file name must include the correct extension (.TTF).

---

## CHAPTER 4 ADMINISTRATION

---

This chapter describes features that are available only if you log into the JasperReports Server as an administrator. Administrators can access the following capabilities of the server that are unavailable to non-administrative users:

- Create, modify, and delete users
- Change a user's password
- Set access permissions on repository folders and objects
- Create, modify, assign, and delete roles

In this chapter, you learn how to perform these tasks.

This chapter contains the following sections:

- [Managing JasperReports Server](#)
- [Managing Users](#)
- [Managing Roles](#)
- [Access Control](#)
- [Configuring Password Options](#)

### 4.1 Managing JasperReports Server

After logging in as an administrator, the main navigation bar includes the Manage menu. The **Manage** menu contains the menu items for managing the server: setting up users and roles, the OLAP Engine, and log levels.

### 4.2 Managing Users

The default installation of JasperReports Server sets up the following users:

| Users in a Default JasperReports Server Installation |                  |                                              |
|------------------------------------------------------|------------------|----------------------------------------------|
| User ID                                              | Default Password | Description                                  |
| anonymousUser                                        | anonymoususer    | Allows anonymous login; disabled by default. |
| jasperadmin                                          | jasperadmin      | Default administrator                        |
| joeuser                                              | joeuser          | Default end user                             |

For more information about anonymous login, see the *JasperReports Server Ultimate Guide*. Initially, you log in as the default administrator, `jasperadmin`, to set up your user account and assign the administrator role to your account. As administrator, you can manage all users and delegate server administration duties by assigning the administrator role to other users. If you assign the administrator role to a user, that user can log into the server as an administrator.

#### To view users:

1. Log in as `jasperadmin`.
2. Select **Manage > Users**.

The **Users** panel lists the user IDs and names in alphabetical order.

3. To display user properties, select a user in the **Users** panel. The user's properties appear in the **Properties** panel.

User properties include:

- User's name – The full name of the user.
- ID – The unique login name for the user, which cannot be changed, stored in the repository.
- Email address – The email address of the user.
- Assigned roles – Hyperlinked roles assigned to the user. Click to open the **Roles** panel for managing the role.
- Status – The enabled or disabled state of the user. Check Status to enable the user to log in, and uncheck to disable the user from logging in.
- Profile attributes – User attributes added directly to the database instead of through the web interface of the server. For more information about user attributes, see the *JasperReports Server Ultimate Guide*.

To filter the list of users or find a specific user, enter a search string in the search field of the Users panel and click



The search results appear in the Users panel, listing all the users whose user name or user ID contains the search string.

#### To create a user:

1. Assuming you are logged in to the server as an administrator, click **Manage > Users**.
2. In the **Users** panel, click **Add User**.
3. In the **Add User** dialog, enter the information shown in [Figure 4-1](#).

**Add User**

User name:  
A McCarthy

User ID (required):  
A\_McCarthy  
Once created, this value cannot be changed.

Email:  
amccarthy@abcfinance.com

Password (required):  
••••

Confirm Password (required):  
••••

☒ User is enabled.

**Add User** **Cancel**

**Figure 4-1 The Add User Dialog**

You can enter the following information when adding a user:

- The user's name, or other identifier, such as a title in any format. Optional. This name appears in the top right-hand corner of the screen when the user logs in.

- The user **ID**, used by the server to identify the user. User IDs must be unique. Required.
- Password and confirmation. Enter the same password in both fields. Required. Passwords are case-sensitive.



Advise users to change their passwords regularly.

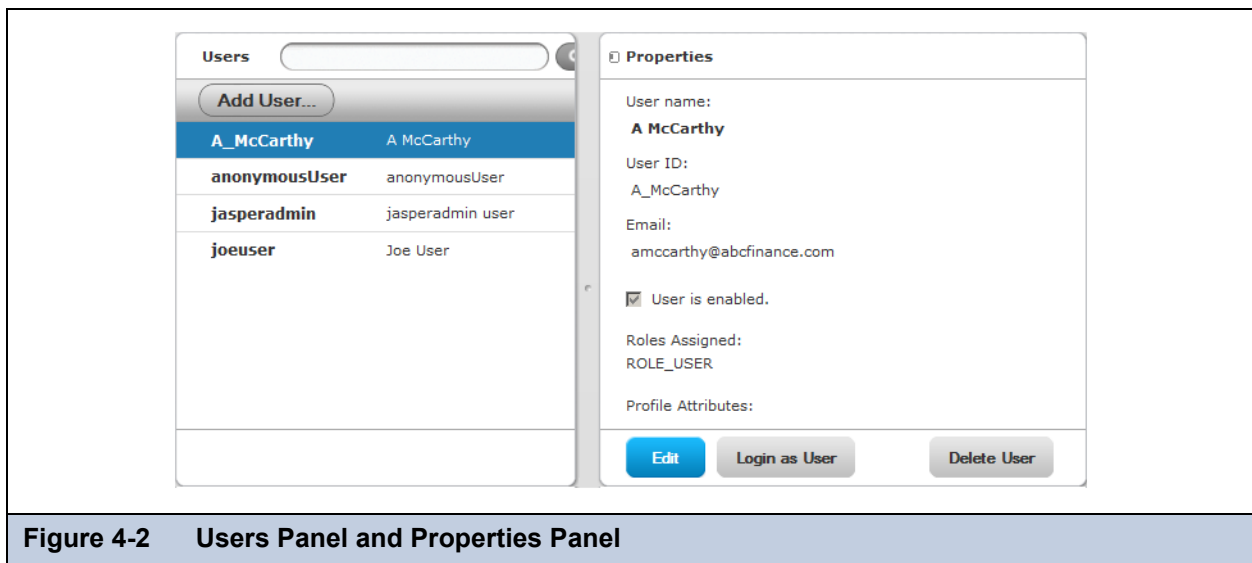
4. Enter the email address of the user in the following format:

username@domain\_name

The email address is optional information in the **Add User** dialog.

5. Check **User is enabled** to enable the user immediately. Uncheck **User is enabled** to prevent a user from logging in. You might want to prevent a user from logging in until you assign custom roles, for example. For more information about roles, see section 4.3, “**Managing Roles,**” on page 71.
6. Click **Add User**, or **Cancel** to quit without saving.

The new user appears in the **Users** panel (unless you used a search term that excludes it). The user’s properties appear in the **Properties** panel. JasperReports Server automatically assigns every user the default role, `ROLE_USER`.

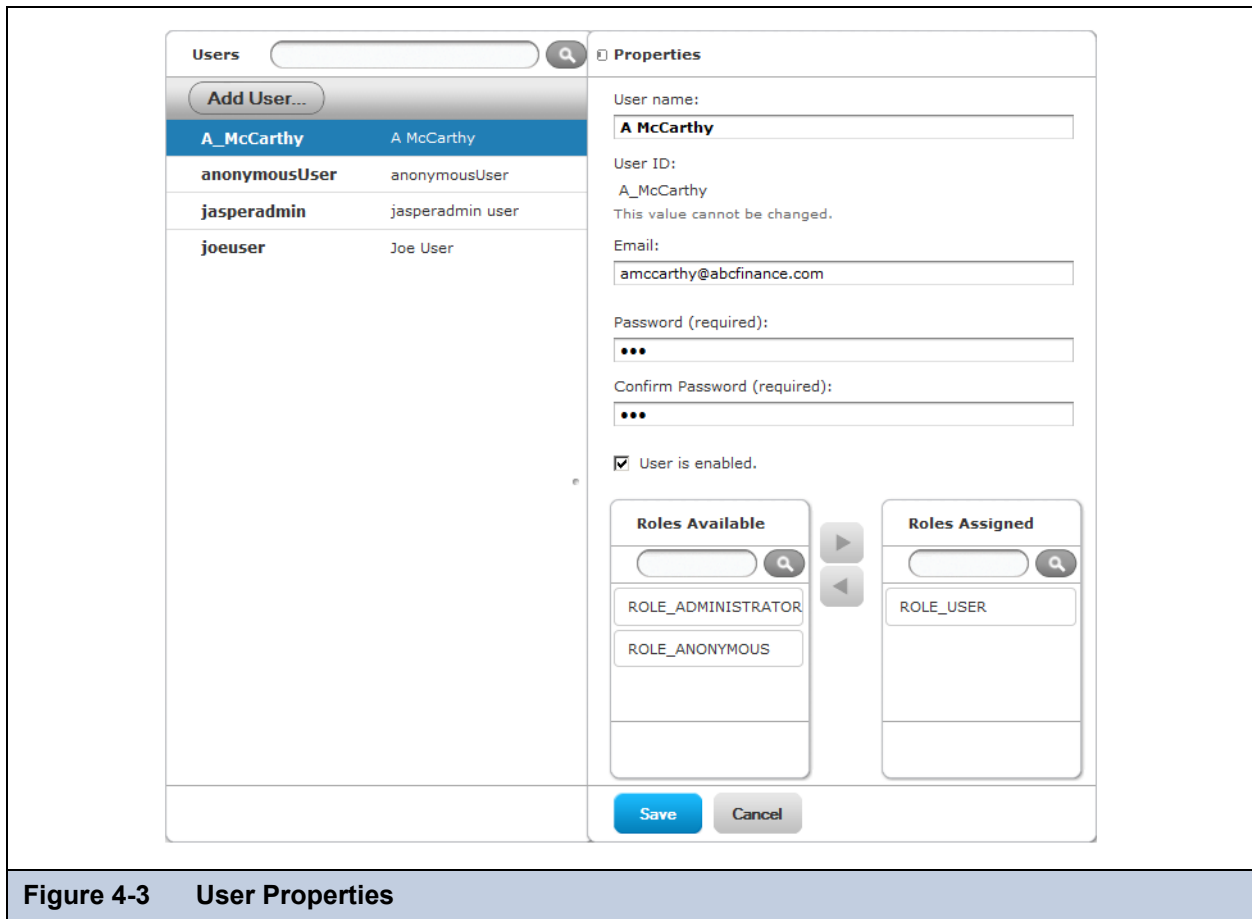


**Figure 4-2 Users Panel and Properties Panel**

#### To edit a user:

1. Assuming you are logged in to the server as an administrator, click **Manage > Users**.
2. In the **Users** panel, select a user. Search by name or ID to find the user if necessary. User properties appear.

3. In the **Properties** panel, click **Edit**. You can now edit user properties other than the user ID, which is read-only, and the profile attributes. Profile attributes must be modified in the database (see the *JasperReports Server Ultimate Guide*).



**Figure 4-3 User Properties**

4. In the Properties panel, modify the user name, email address, or password.



Although you can change a user's password, even as Administrator, you cannot read the existing password in clear text.

5. To assign the user one or more new roles:
  - a. In the **Roles Available** pane, select one or more roles.




Use Ctrl-click to select multiple roles.

- b. Click  to move roles to the **Roles Assigned** pane. For example, in **Roles Available**, select

ROLE\_ADMINISTRATOR, and click



6. To remove roles from the user, select the roles, and click  to move the roles into **Roles Available**.

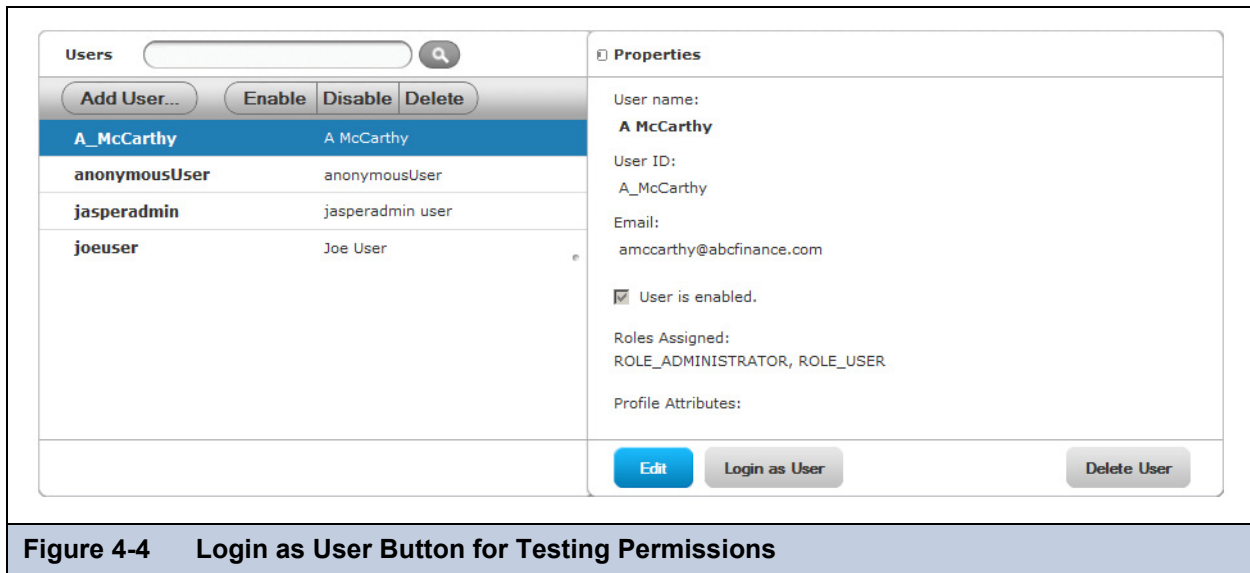


Roles Assigned lists the roles currently assigned to the user. Roles Available lists unassigned roles for which the user is eligible.

For more information about creating and adding roles, see section 4.3, “Managing Roles,” on page 71.

7. To enable or disable the selected user, check or uncheck the **User is enabled** check box.
8. Click **Save**.

The Properties panel lists the updated roles for the selected user in **Roles Assigned**.



**Figure 4-4 Login as User Button for Testing Permissions**

9. In the Properties panel, click **Login as User** to test the user's permissions.

#### To delete a user:

1. Assuming you are logged in to the server as an administrator, click **Manage > Users**.
2. Select a user.  
Properties of the user appear in the Properties panel.
3. In the **Users** panel, click **Delete**, or in the Properties panel, click **Delete User**.  
The user is removed from JasperReports Server.

#### To delete, enable, or disable multiple users:

1. Assuming you are logged in to the server as an administrator, click **Manage > Users**.
2. In the Users panel, select each user, one at a time, and click the **Enable**, **Disable**, or **Delete** button.

## 4.3 Managing Roles

Roles define sets of users who are granted similar permissions. Administrators create roles, assign roles, and grant permissions to users. For more information about assigning permissions, see section 4.4, “Access Control,” on page 75.

The default installation of JasperReports Server sets up the following roles:

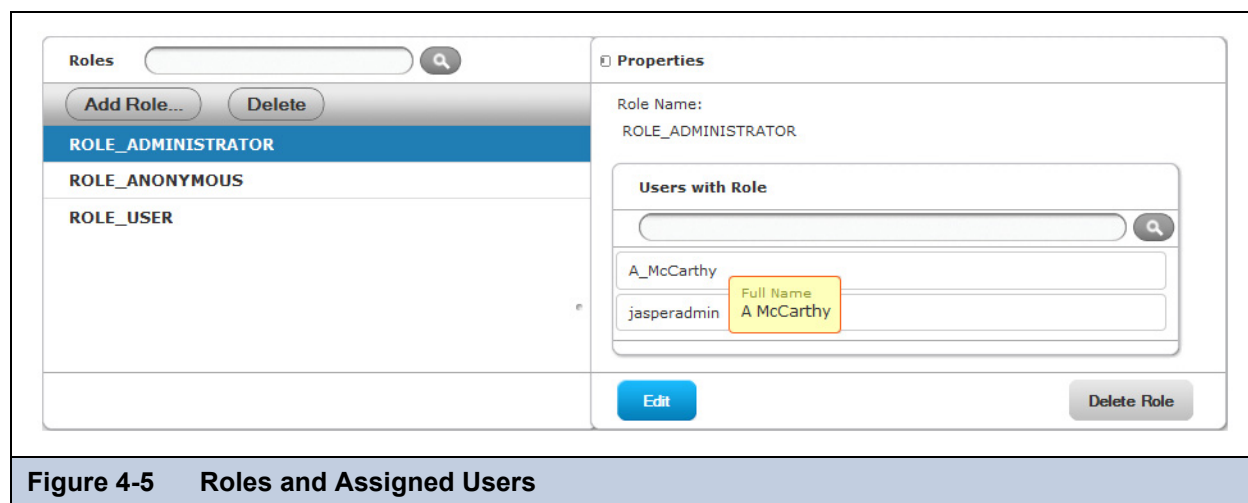
| Roles in Default JasperReports Server Installation |                                                                                                    |
|----------------------------------------------------|----------------------------------------------------------------------------------------------------|
| Role                                               | Description                                                                                        |
| ROLE_ADMINISTRATOR                                 | A special system-level role for creating users and delegating administration privileges to others. |

| Roles in Default JasperReports Server Installation, continued |                                                                                                                                                                                                                                                                        |
|---------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Role                                                          | Description                                                                                                                                                                                                                                                            |
| ROLE_USER                                                     | A role that the server automatically assigns to every user that is created. This role cannot be removed.                                                                                                                                                               |
| ROLE_ANONYMOUS                                                | A role that the server automatically assigns to any agent accessing the server without logging in, providing anonymous access is enabled. This role is also assigned to the default anonymous user. By default, anonymous access is disabled and this role isn't used. |

Administrators manage all roles they create, but not those created by other administrators. The server enforces this scheme to insure that only valid roles are assigned to users.

#### To view roles:

1. Log into the server as an administrator, and select **Manage > Roles**.  
The **Roles** panel lists the existing roles.
2. In the **Roles** panel, click a role to view the users assigned to the role.  
The **Properties** panel lists the users assigned to the selected role.
3. In **Users with Role**, point the mouse at the user **ID**. The full name of the user appears in the tool tip.



**Figure 4-5 Roles and Assigned Users**

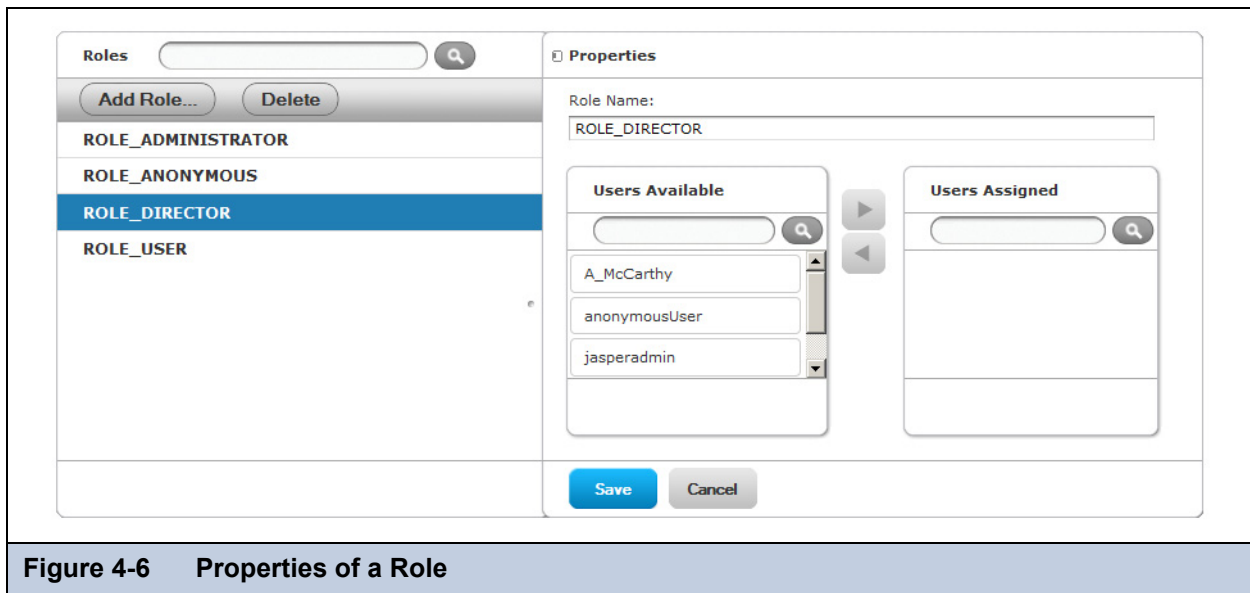
#### To create a role:

1. Assuming you are logged in to the server as an administrator, click **Manage > Roles**.
2. Click the **Add Role** menu item.  
The Add Role dialog appears.
3. Enter the name of the role. For example, enter `ROLE_DIRECTOR`.  
A warning appears if you enter a role name that is not unique.
4. Click the **Add Role** button to create the role, or click **Cancel** to quit without saving.  
No users have been assigned the role yet.

#### To assign a user to a role:

1. Assuming you are logged in to the server as an administrator, click **Manage > Roles**.
2. Select a role, for example `ROLE_DIRECTOR`, in the **Roles** panel.
3. In the **Properties** panel, click **Edit**.  
The information in the Properties panel becomes editable. The available users and those already assigned to the selected role appear, as shown in [Figure 4-6](#).





**Figure 4-6 Properties of a Role**

The **Users Assigned** panel lists the user IDs of users currently assigned to the role. The **Users Available** panel lists the user IDs of users to whom you can assign the role. If the list of user IDs is long, use the search field to find specific user IDs, and use the tool tips to see their names.

You can change the role name and assign users to and remove users from the role.



Changing the name of the role affects all users who are assigned the role. The role name associated with permissions in the repository is also updated. Changing the role name may compromise permissions defined in security files for analysis. For more information, see the *Jaspersoft OLAP User Guide*.

4. Select the user in **Users Available**. To assign multiple users to a role, use Ctrl-click to select them.



5. Use to move one or more users into **Users Assigned**.

#### To remove users from a role:

1. Select the user to remove in **Users Assigned**. To remove multiple users from a role, use Ctrl-click to select them.



2. Use to move one or more users into **Users Available**.
3. Click **Save** to keep your changes, or **Cancel** to quit without saving.




To assign a user to multiple roles, select each role, one at a time, and assign it to the user. Alternatively, edit user properties as described in [To edit a user: on page 69](#).

#### To delete a role:

1. Assuming you are logged in to the server as an administrator, click **Manage > Roles**.
2. Select the role in the **Roles** panel.
3. In the **Roles** panel, click **Delete**, or in the **Properties** panel, click **Delete Role**.  
The role is removed completely from the server.

#### To filter a list of roles:

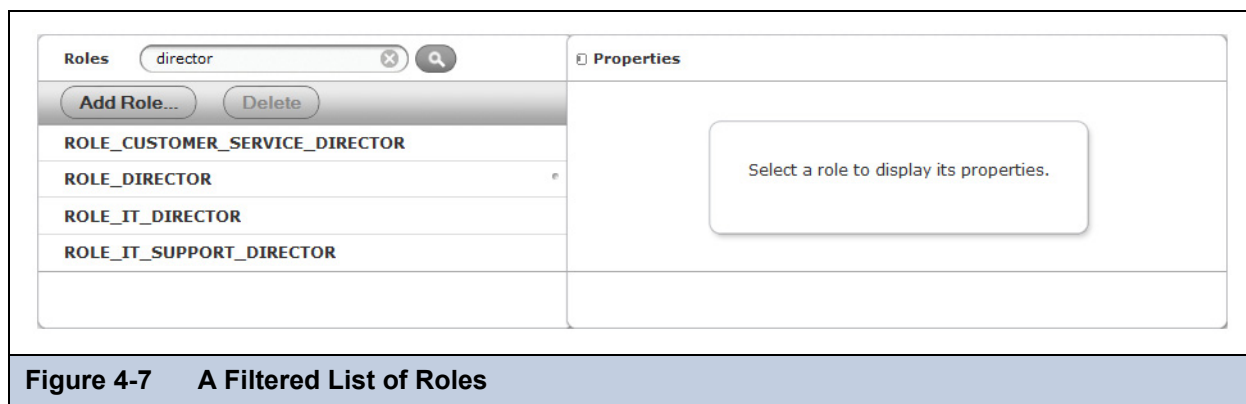
1. Assuming you are logged in to the server as an administrator, click **Manage > Roles**.  
The **Roles** panel lists the existing roles.
2. To find all roles of a certain type, which you can identify by role name, use a filter. For example, find all director-level roles.

- a. Enter a search string, for example `director`, in the search field of the **Roles** panel.
- b. Click .



Search strings are not case-sensitive.

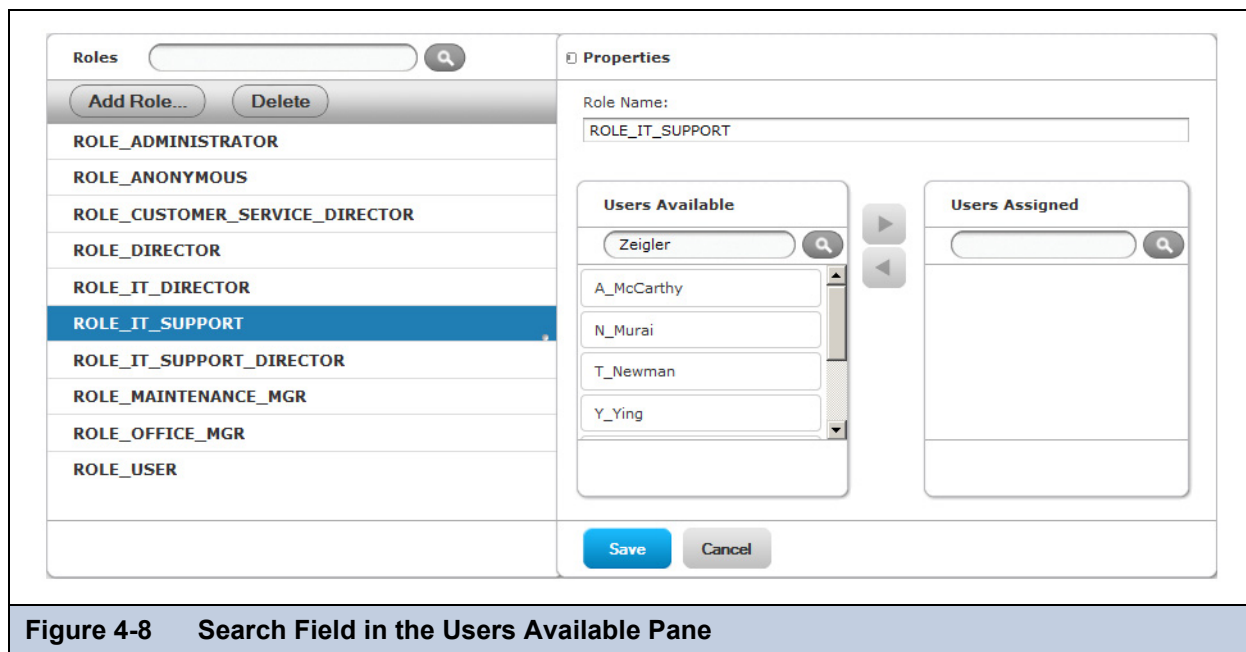
The filtered list only includes roles that match the search string, so it's easier to manage.



**Figure 4-7 A Filtered List of Roles**

**To filter the users available for or assigned to roles:**

1. Assuming you are logged in to the server as an administrator, click **Manage > Roles**.  
The Roles panel lists the existing roles.
2. In **Roles**, select a role.
3. In **Properties**, select **Edit**.  
The users available for and assigned to roles appear in **Properties**.
4. In the Users Available or the Users Assigned search field, enter a search term. For example, enter Zeigler in the **Users Available** search field.



**Figure 4-8 Search Field in the Users Available Pane**

5. In **Users Available**, click .

The user ID of any users matching the search term, Zeigler, appears in Users Available.

## 4.4 Access Control

Access control ensures that users of JasperReports Server access only the data they are authorized to see. The server provides access control through an integrated security framework that includes:

- Authentication – Restricts access to identified users and protects against unauthorized access by requiring passwords. Defines roles for grouping users and assigning permissions.
- Authorization – Controls access to repository objects, pages, and menus based on users and roles.

### 4.4.1 Authentication Overview

The first security-related task is to define user accounts and secure them with passwords. Users must log in with their user ID and password to use JasperReports Server. The server stores user definitions, including encrypted passwords, in a private database. Administrators create, modify, and delete user accounts using the **Manage** menu, as described in section 4.2, “Managing Users,” on page 67.

Roles can be assigned to any number of users. Using roles, administrators create groups or levels of users that are granted certain permissions. For example, administrator privileges are granted by the role, `ROLE_ADMINISTRATOR`. A user may belong to any number of roles and receive the privileges from each of them. The server stores role definitions in its private database, and administrators create, modify, and delete roles through the administrator pages, as described in section 4.3, “Managing Roles,” on page 71.

JasperReports Server relies on the open source Acegi security framework; it has many configurable options for:

- External authentication services such as LDAP (used by Microsoft Active Directory and Novell eDirectory)
- Single sign-on using JA-SIG's Central Authentication Service (CAS)
- Java Authentication and Authorization Service (JAAS)
- Container security (Tomcat, Jetty)
- SiteMinder
- Anonymous user access (disabled by default)

JasperReports Server also supports these encryption and authentication standards:

- HTTPS, including mandatory HTTPS
- HTTP Basic
- HTTP Digest
- X509

The Acegi framework is readily extensible to integrate with custom and commercial authentication services and transports.

Authentication occurs by default in one of or both of the following ways:

- Through the web user interface, forcing login
- Through HTTP Basic authentication for web services and for XML/A traffic

The server can automatically synchronize with an external authentication service. The external users don't need to be created manually in the server first. Both users and roles are created automatically in the server from their definitions in an external authentication service. For an overview of the authentication system and details about external authentication, see the *External Authentication Cookbook*.

### 4.4.2 Authorization Overview

With a user's identity and roles established, JasperReports Server controls the user's access in these ways:

|                        |                                                                                                                                                                                                                                                                                                                                                                                                           |
|------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Menu options and pages | The menus that appear in JasperReports Server depend on the user's roles. For example, only users with the administrator role can see the <b>Manage</b> menu. By modifying the server's configuration, you can modify access to menus, menu items, and individual pages. Refer to the <i>JasperReports Server Source Build Guide</i> and <i>JasperReports Server Ultimate Guide</i> for more information. |
|------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

|                                     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|-------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Resource permissions                | Administrators can define access permissions on every folder and resource in the repository. Permissions are defined for every role and every user, or left undefined to be inherited from the parent folder. For example, users may have read-write access to a folder where they create reports, but the administrator can also create shared reports in the same folder that are set to read-only. The possible permissions are no access, execute only, read-only, read-delete, read-write-delete, and administer (see section 4.4.3, “Permissions,” on page 76).                       |
| Resource permissions<br>(continued) | Permissions are enforced when accessing any resource either directly through the repository interface, indirectly when called from a report, or programmatically through the web services.                                                                                                                                                                                                                                                                                                                                                                                                  |
| Administrator privileges            | JasperReports Server distinguishes between reading or writing a resource in the repository and viewing or editing the internal definition of a resource. For security purposes, granting a user read or write permission on a resource doesn’t allow viewing or editing the resource definition. For example, users need execute or read permission on a data source to run reports that use it, but they cannot view the data source’s definition which includes a database password. Also, only administrators may interact with theme folders to upload, download, and make them active. |
| Data-level security                 | <p>Data-level security defines what data can be retrieved and viewed in a report, based on the user ID and roles of the user who runs the report. For example, a management report could allow any user to see the management hierarchy, managers to see the salary information for their direct employees, and only human resource managers to see all salary information.</p> <p>Data-level security through analysis views is covered in the <i>Jaspersoft OLAP User Guide</i>.</p>                                                                                                      |
| Profile attributes                  | A profile attribute is a name-value pair inserted in a user object, such as a user account or role. The attribute extends the object’s standard access grants. It can be added only by users with administrator privileges. For information about defining profile attributes, see the <i>JasperReports Server Ultimate Guide</i> .                                                                                                                                                                                                                                                         |

### 4.4.3 Permissions

Permissions on folders and resources determine what users see in the repository and what actions they are allowed to perform. In the following table, the actions granted for each permission include all of the actions granted for permissions above it, except for the No Access permission. The actions granted for each permission strictly exclude all of the actions granted for permissions below it.

| Permission   | Actions Granted on Repository Folders and Resources                                                                                                                                                                                                                                                                                                                                                                   |
|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| No Access    | Users can never see or access the folder or resource either directly in the repository or indirectly when running a report or OLAP view.                                                                                                                                                                                                                                                                              |
| Execute Only | Users can never see the folder or resource in the repository, but the reports or OLAP views that they run can access them.                                                                                                                                                                                                                                                                                            |
| Read Only    | <ul style="list-style-type: none"> <li>• See the folder or resource in any JasperReports Server dialog</li> <li>• See the properties of a folder or a resource</li> <li>• Copy a folder and all of its readable contents</li> <li>• Copy resources individually or in bulk</li> <li>• View (run) a report or OLAP view</li> <li>• Run a report in the background</li> <li>• Schedule a report to run later</li> </ul> |

| Permission                        | Actions Granted on Repository Folders and Resources                                                                                                                                                                                                                                                                                                                                                                              |
|-----------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Read + Delete                     | <ul style="list-style-type: none"> <li>• Cut (move) a folder and all of its contents</li> <li>• Delete a folder and all of its contents</li> <li>• Cut (move) resources individually or in bulk</li> <li>• Delete resources individually or in bulk</li> </ul>                                                                                                                                                                   |
| Read + Write + Delete             | <ul style="list-style-type: none"> <li>• Add a subfolder</li> <li>• Paste into a folder (copy or cut)</li> <li>• Save the output of a scheduled report in a folder</li> <li>• Rename a folder or resource and change its description string</li> <li>• Add a JasperReport resource to the repository (upload a JRXML)</li> <li>• Edit the definition of a JasperReport resource in the repository (replace the JRXML)</li> </ul> |
| Administer                        | <ul style="list-style-type: none"> <li>• Set the permissions (by role and by user) on a folder or resource. This effectively delegates certain repository administration tasks.</li> </ul>                                                                                                                                                                                                                                       |
| Administer and ROLE_ADMINISTRATOR | <ul style="list-style-type: none"> <li>• Add (create) a resource in a folder</li> <li>• Edit a resource, for example the components of a report unit</li> </ul>                                                                                                                                                                                                                                                                  |

Permissions apply when browsing or searching the repository, as well as when using any dialog that accesses the repository, such as when browsing folders to save a report. Note that:

- Copying does *not* preserve the permissions on an object. Users may copy a read-only object, paste it into a read-write folder, then edit the object.
- Copying and cutting (moving) actions can only be completed if the user has Read + Write + Delete access to the folder in which the object is pasted. For more details, see section 1.5, “Moving Folders,” on page 13.
- Cutting, deleting, and setting permissions on folders is allowed only if the user has the same permission on all folder contents. Cutting and deleting resources in bulk is allowed only if the user has at least Read + Delete permission on all selected resources.
- Deleting a resource or the contents of a folder is allowed only if no other resources rely on them.

#### 4.4.3.1 Inherited Permissions

According to the permission architecture, there is a permission setting for every user and role on every folder and resource in the repository. To simplify the definition of permissions, JasperReports Server supports the inheritance of permissions from the parent folder of a folder or resource. If permissions on a given folder or resource are not explicitly set for a user or role, the user or role has the same access permission on that folder or resource as they have on the parent folder. When a permission is explicitly set on a folder or resource, that permission overrides inherited permissions on the parent folder.

Using this mechanism, administrators can manage large hierarchies of content and keep them secure. When the administrator explicitly sets a permissions on a folder for a user or role, the folder’s contents and its subfolders recursively inherit the permissions, unless the subfolders or contents have an explicit setting of their own.

#### 4.4.3.2 Cumulative Permissions

Because permissions can be assigned to both users and roles, a user belonging to one or more roles may have multiple permissions explicitly set or inherited on any given folder or resource. In fact, every permission must be defined on the root, even if it has the default value of No Access, and therefore every role- and user-based permission on every folder and resource has a setting through inheritance. Therefore, for every folder or resource, every user has their own user-based permission and the permission assigned to the `ROLE_USER`.

How does the server determine the effective permission from the many that apply? Permissions in the server are strictly cumulative, meaning that the least restrictive among the set of all permission applies. Even if a more restrictive permission, such as No Access, is set explicitly, the less restrictive permission, such as Read-Only applies, regardless of whether it is inherited or set explicitly.

#### 4.4.3.3 Execute-Only Permission

This release of JasperReports Server introduces execute-only permission to the repository. As in file systems, execute-only permission allows running reports and OLAP views to access a resource, but keeps the resource from appearing in the repository.

Execute-only permission applies to folders as well, keeping them from appearing in folder tree when users browse the repository, yet still allowing the resources they contain to inherit the execute-only permission. This is useful for hiding folders and resources such as data sources that only administrators and data analyst roles need to access in the repository. However, if your execute-only folder contains read-only resources, those resource are hidden when you browse folders, but can be found, either accidentally or intentionally, when you search the repository.

Like other permissions, execute-only permission is either role-based or user-based; consequently, certain users may access a resource from a running report, but other users cannot.



If you have data or sensitive content in a resource, set the permissions for users or roles that you want to deny access to No Access.

Hiding a resource with execute-only permission doesn't protect it against access, because malicious users who find the resource ID may be able to create a report or OLAP view that extracts the sensitive content.

#### 4.4.3.4 Default User Permissions

For non-administrator users, the default permission at the root is No Access and any permissions must be explicitly defined. In practice, the default installation of the repository contains sample data with a mix of No Access, Execute-only, Read-only, and Read-write permissions that allow the sample users to access folders and resources. The sample permissions demonstrate a common approach to permissions, allowing users to see the resources they can access and hiding the ones they can't, while administrators have full access.

We recommend you familiarize yourself with the permissions mechanism by viewing and setting permissions in the sample data, as described in the following section.

### 4.4.4 Assigning Permissions

Administrators can assign permissions to access any folder or resource in the repository. Users with the Administer permission on a folder can assign permissions to that folder and any contents that inherit the permission. Users granted Administer permission on a resource can set only the permissions on that specific resource.

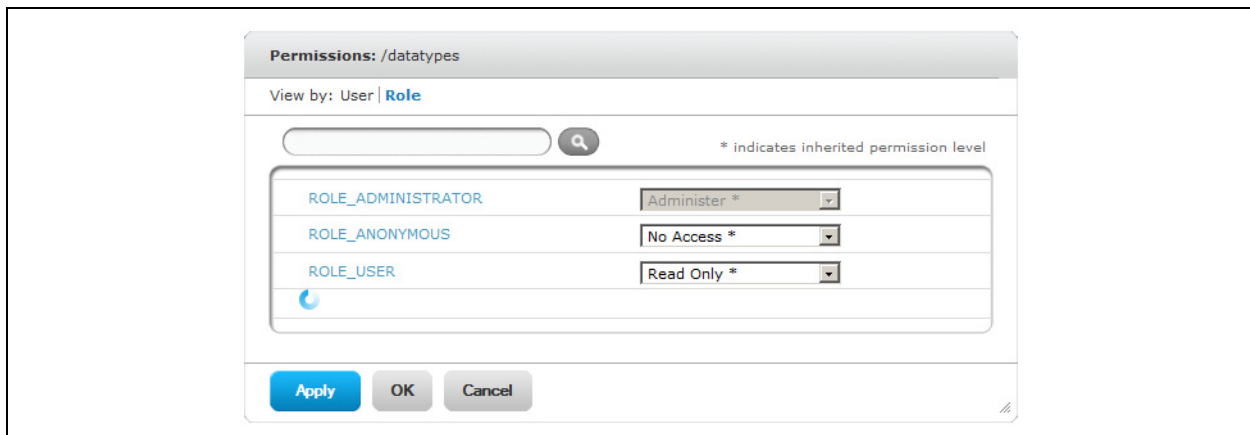


The examples in this section assume that only the roles and users set up by the default installation are in place.

#### To set permissions on a folder or resource in the repository:

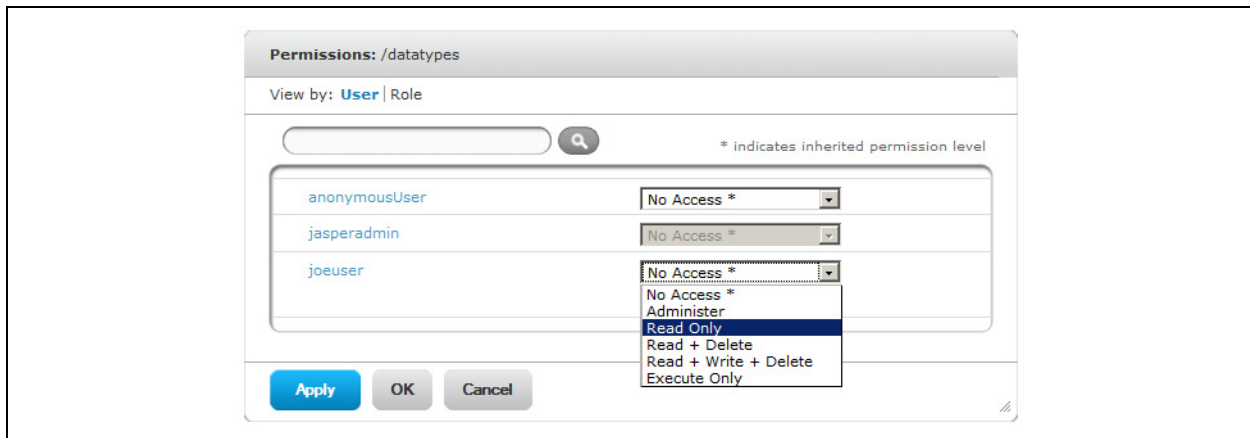
1. Log into the server as an administrator.
2. In the repository, right-click a folder or resource. For example, right-click the sample Input Data Types folder.
3. Select **Permissions** from the context menu.

The **Permissions** dialog opens, showing the permissions in effect for the selected object, the Input Data Types folder. By default, the **View by** option is set to role. An asterisk (\*) means that the permission is inherited from the object's parent.



**Figure 4-9 Permissions Window, View by Role**

4. In the Permissions dialog, change the **View by** option. Click **User**.  
The permissions assigned to users appear.
5. Select a permission for a user from the drop-down. For example, select Read Only for joeuser.



**Figure 4-10 Permissions Window, View by User**

**Figure 4-10** shows the default permissions on the folder by user: No Access. The default installation assigns permissions only to **ROLE\_ADMINISTRATOR** and **ROLE\_USER** and assigns no permissions to users.



You cannot change the jasperadmin user permissions, as indicated by the unavailable drop-down in **Figure 4-10**, nor can you change the permissions of any other administrator or the **ROLE\_ADMINISTRATOR**. This prevents administrators from inadvertently removing their ability to set permissions.

6. Click **Apply** to grant read-only permission to joeuser.



You must apply the permission setting before toggling the **View by** setting in the next step; otherwise, any changes to permission settings are ignored.

Read-only permission gives the joeuser the ability to see, but not modify, the Input Data Types folder and its contents. For other end users, whose permissions are defined only by **ROLE\_USER**, access to the folder is still execute-only.

7. Click **Role** to toggle the **View by** setting.  
The permissions assigned to roles reappear.  
You can select a permission from the drop-down, and click **Apply** to set a permission for the role.
8. Click **OK** to return to the repository view.



There are two special cases when setting permissions:

- If a resource inherits a permission, for example Read-Only, you cannot set the permission to the same value, at least not directly. You need to temporarily change the permission level on the parent folder, then set the explicit permission, then set the parent folder's permission back to the original value.



When a resource and its parent folder have been set to the same permission in this way, the permission dialog still shows the asterisk as if the permission were inherited. But when the parent is later given a different permission, for example Read-Write, the resource will retain its explicit Read-Only permission instead of inheriting Read-Write.

- To reset the permission level so that it once more inherits from its parent folder, select a different permissions level and click Apply, then select the permission with the asterisk and click Apply again.

### 4.4.5 Testing User Permissions

Once you have configured users, roles, and permissions on repository objects, Jaspersoft recommends that you test the permissions granted to a few representative users. Also test permissions when you add new users, roles, and resources, and when you make any major modifications to your access control configuration.

#### To test user permissions:

1. Log into the server as an administrator.
2. Select **Manage > Users**, and in the Users panel, select the user whose access permissions you want to test.
3. In the Properties panel, click **Login as User**.  
The selected user's Home page appears. The login information in the upper-right corner shows that you are logged in as that user.
4. In the repository, browse or search for the folders and resources to test for user access.
5. Verify that JasperReports Server can access the expected folders and resources. Make a note of any objects that should be accessed but are not, and any objects that should be hidden but are displayed.
6. Click **Log Out**.  
The manage users page reappears.
7. To change the user's permissions, perform one or more of the following tasks:
  - Edit object permissions in the repository.
  - Modify users.
  - Modify roles.
8. Continue testing until you verify that the user's permissions are correct.
9. Repeat these steps with several representative users to ensure that your access control is properly configured; an access control configuration that hasn't been tested probably doesn't secure your data adequately.

## 4.5 Configuring Password Options



By default, passwords are stored in an encrypted format in the repository database. For information about how to change the way passwords are encrypted, refer to the *JasperReports Server Installation Guide*.

### 4.5.1 Enabling Password Expiration



If your users are externally-authenticated, do not enable this option.

If your security policies require users to change their passwords at regular intervals, you can enable password expiration. In this case, JasperReports Server prompts users to change their passwords at the interval you specify. For example, if you set the password expiry to 90 days, the server prompts your users to change their passwords every three months. When a user's password expires, the user cannot log in until she changes her password. The default value is 0; in this case, passwords don't



expire. When the password expiration option is enabled, the server automatically enables the **Change password** link on the Login page, even if `allowUserPasswordChange` is set to false.

| Password Administration Option                                                                                                                |                                  |                                                                                                             |
|-----------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------|-------------------------------------------------------------------------------------------------------------|
| Configuration File                                                                                                                            |                                  |                                                                                                             |
| ...\\WEB-INF\\jasperserver-servlet.xml (controls the Login page)<br>...\\WEB-INF\\applicationContext-security-web.xml (controls web services) |                                  |                                                                                                             |
| Property                                                                                                                                      | Value                            | Description                                                                                                 |
| <code>passwordExpirationInDays</code>                                                                                                         | 0 <default><br><any other value> | Set the value to any positive, non-zero value to specify the number of days after which a password expires. |

### 4.5.2 Allowing Users to Change their Passwords

The password administration option controls whether or not the **Change Password** link appears on the Login page. In order to change their passwords, users click the **Change password** link. Enable this link only if JasperReports Server authenticates your users. If your users are externally-authenticated, do not enable the option.

To enable the password administration option that allows users to change their passwords, edit the configuration file as described in the following table.



Enabling the password expiration option (described in the previous section) automatically enables the capability on the Login page for changing passwords. The **Change password** link appears above the Login button.

| Password Administration Option         |                         |                                                                                                            |
|----------------------------------------|-------------------------|------------------------------------------------------------------------------------------------------------|
| Configuration File                     |                         |                                                                                                            |
| ...\\WEB-INF\\jasperserver-servlet.xml |                         |                                                                                                            |
| Property                               | Value                   | Description                                                                                                |
| <code>allowUserPasswordChange</code>   | false <default><br>true | Set the value to <code>true</code> to enable the <b>Change password</b> link. Any other value disables it. |

Restart the `jasperreportsTomcat` service after changing the servlet.



## CHAPTER 5 THEMES

---

JasperReports Server 4.0 introduces a new look and feel, and more importantly, a complete mechanism to easily modify the appearance of the user interface (UI). This mechanism called *themes* is based on Cascading Style Sheets (CSS), the web standard for defining the appearance of HTML content.

The set of all CSS files and associated images that define the appearance of the user interface is a theme. Themes are stored as file resources in folders in the repository, with special menus on theme folders for activating, uploading, and downloading a theme. You can store any number of themes in the repository, and administrators can switch between them, providing an easy and quick way to change the default appearance to suit your needs.

This chapter contains the following sections:

- **Introduction**
- **How Themes Work**
- **Administering Themes**
- **Working With CSS Files**

### 5.1 Introduction

The JasperReports Server user interface is based on CSS (Cascading Style Sheets) files that define the styles of the elements appearing in the HTML, itself defined in and generated from JSP (JavaServer Pages) and JavaScript. A theme is a collection of CSS files and associated images that specify the appearance for all or part of the user interface. A theme only controls how the interface appears, for example fonts, colors, spacing, lines, and image elements of the UI. It does not control what appears, such as the contents of menus or the effect of clicking a button.

Themes are defined globally at the repository root, so every user sees the same theme. Only administrators can set the theme. Administrators can add, upload, edit, copy, and delete the files that make up the theme, just like other resources in the repository. The repository provides special actions on theme folders for downloading and uploading themes as ZIP (compressed archive) files, and for activating the theme.

The themes mechanism is very flexible, allowing administrators to easily change the global appearance. For example, the following scenarios are possible with the new themes mechanism:

| Scenario                          | Description                                                                                                                                                                                                                                                          |
|-----------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Use the default theme unchanged.  | The new UI has been designed for clarity and ease-of-use. After a standard installation, the default theme is set and every user sees the server with this interface. If the default theme suits your needs, there is no need to customize it or develop new themes. |
| Quickly modify the default theme. | You can specify overrides of individual CSS rules or replace images in the default theme. It is easy to create or upload the new files and activate your customizations.                                                                                             |
| Create an entirely new theme.     | With CSS experience, you can change the entire look and feel of the server. The server UI can be tailored to match or blend in with nearly any other web design.                                                                                                     |

It is important to realize that a theme refers to two concepts simultaneously:

- A folder containing a set of CSS files and image files in the proper location in the repository.
- The entire appearance of the user interface after activating the theme's files. However, most themes only define a few files and rely on the inheritance mechanism to load the other files from the default theme. In fact, except for the default theme, the entire user interface is rarely defined in a single theme.

For example, a very simple theme named MyLogo contains a custom image file to replace the Jaspersoft logo, and nothing else. The rest of the interface is inherited from the default theme. Yet we say that MyLogo is the active theme, and every user sees the MyLogo theme.

## 5.2 How Themes Work

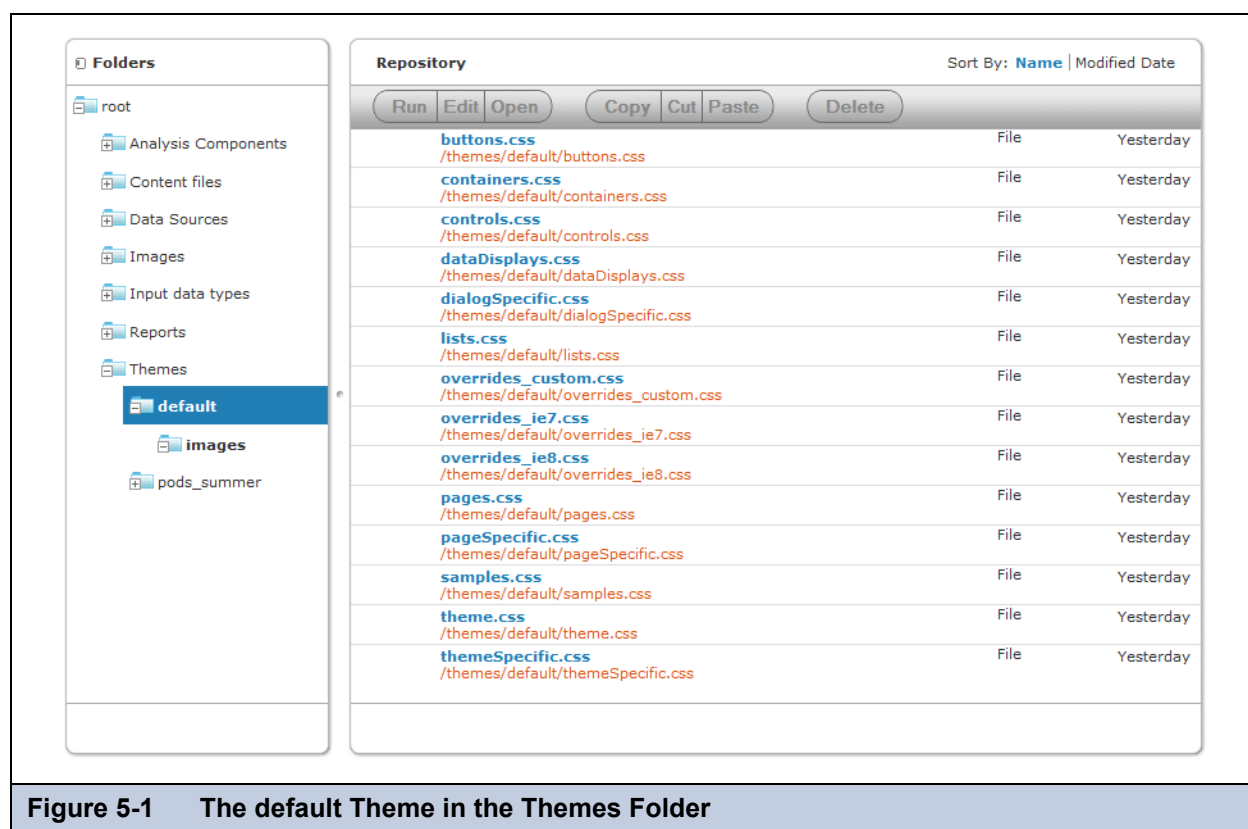
Themes are stored in a special folder named Themes that appears at the root of the repository. The Themes folder contains a default theme and any number of custom theme folders. Each theme is stored in its own folder and is known by the name of the folder. The folder named “default” contains the complete definition of every style that makes up the default theme shipped with JasperReports Server. The default theme cannot be modified, even by administrators.



You cannot modify the files of the default theme through the repository. If you try to do so by circumventing the repository, you could inadvertently change rules that make the UI unusable. In this situation you would need to re-install JasperReports Server to recover.

The Themes folder has execute-only permission for all users except administrators. However, only administrators have access to the special context menus for themes, so regular users cannot change themes. To avoid situations that can potentially make the UI inaccessible, do not change the permissions on the Themes folder.

In the Themes folder, one of the themes is designated as the active theme; the name of the folder (and its subfolders) are bold in the repository to indicate that it is the active theme. All users will see the active theme on all pages, including the login page. After installation, the default theme is the active theme. The following figure shows the CSS files of the default theme.



**Figure 5-1 The default Theme in the Themes Folder**

### 5.2.1 Theme Files

A complete theme consists of exactly the following files, along with all referenced images:

- |                   |                     |                          |
|-------------------|---------------------|--------------------------|
| 1. theme.css      | 5. lists.css        | 9. dialogSpecific.css    |
| 2. pages.css      | 6. controls.css     | 10. overrides_ie7.css    |
| 3. containers.css | 7. dataDisplays.css | 11. overrides_ie8.css    |
| 4. buttons.css    | 8. pageSpecific.css | 12. overrides_custom.css |

In addition, the default theme contains the file `samples.css` that is only used by the **View > Samples** page described in section 5.4.4, “User Interface Samples,” on page 91. The files `overrides_ie7.css` and `overrides_ie8.css` are only loaded when the user’s browser is Internet Explorer 7 or 8, respectively.

The default theme stores referenced image files in a folder named `images`. There are approximately 80 image files in the default theme. There are two ways to change an image from the default theme:

- Use a folder named `images` and image files with the same name as the ones you want to replace.
- Modify the corresponding CSS rules to redefine the location (folder and file name) where they can be found.

When you modify the CSS rules, you can use any of the following ways to reference image files, or any other helper file:

- Directly in the theme folder. In this case the file is referenced without a path, for example `"myfile.png"` in CSS.
- In any folder structure located in the theme folder. For example, your custom CSS file could refer to `"MyImages/myfile.png"` if you create a folder named `MyImages` in the theme folder and upload your images there.
- Anywhere on the internet. Following the CSS standard, your custom CSS can refer to images, or any helper file, with a regular URL.

### 5.2.2 Inheritance Mechanism

In order to render the user interface, JasperReports Server must load each of the theme files. Because each file can be stored in multiple themes, the inheritance mechanism determines which file to load.

The server loads each of the CSS files in section [5.2.1, “Theme Files,” on page 85](#) in the order they are listed. To locate the file, the server looks in the following locations, in the following order:

1. The active theme folder.
2. The default theme.

When one of the CSS files references an image file or a helper file, including any path to that file, the server looks for that path and filename in the same two locations, in the same order. In this way, each file and image is resolved first in the active theme, and if not found, then in the default theme. The active theme does not need to contain all the files because the default theme is guaranteed to have them.

### 5.2.3 CSS Priority Scheme and Custom Overrides

Once the inheritance mechanism determines which files to load, the standard CSS priority scheme determines which rules will be visible, based on the order in which files are loaded.

This leads to two general ways of developing custom themes:

- The quickest way is to copy individual CSS rules from the default theme files, modify the rules to change the UI, and save them in the `overrides_custom.css` file. Because `overrides_custom.css` is always the last CSS file to be loaded, its rules will override the same rules in other files. This allows you to easily change any number of rules, and manage them all in a single file.

For example, if you want to increase the size of text on all the buttons in the default theme, you can do this with a few rules in the `overrides_custom.css` file. You may need to adjust the spacing for certain buttons, but the idea is you only need to change a limited number of rules.

- If you modify the user interface extensively, you can use the existing structure of CSS files in the default theme. In this case, copy the relevant files from the default theme, make your modifications, and save the file in your new theme. The inheritance mechanism will use this new file when you activate the theme.

An example of these extensive changes would be if you want to increase the size of the buttons themselves in the default theme. You would need to rewrite the majority of the rules in the `buttons.css` file and create images for the new buttons. In this case, it is much easier to copy the `buttons.css` file than to copy dozens of rules into the `overrides_custom.css` file. You could still use the `overrides_custom.css` file to adjust the spacing of elements around the buttons, because there would be fewer of those rules to modify.

Jaspersoft recommends using the custom overrides method for most custom themes. A theme that changes simple appearances such as colors, fonts, and spacing has relatively few rules and is easily manageable in a single file. And many changes can be made by copying and modifying image files, without writing any CSS rules. Only if you change the fundamental layout or appearance of the user interface should you consider copying and modifying the other CSS files.

If you do make a copy of the other CSS files, your copy of the file must contain all of the CSS rules as the original. If any rules are accidentally deleted or modified, even by a single character, the theme may not work properly.

## 5.3 Administering Themes

Themes are sets of CSS and image files stored in the Themes folder at the root of the repository. In the repository browser, the Themes folder and individual theme folders have special actions for administrators to manage them. You can also use the repository search to find CSS and image files.

The folders and actions for managing themes are visible only to administrators. The Themes folder has execute-only permission for `ROLE_USER` so that all users can load the theme files and see the user interface, but not access the folders and files in the repository.

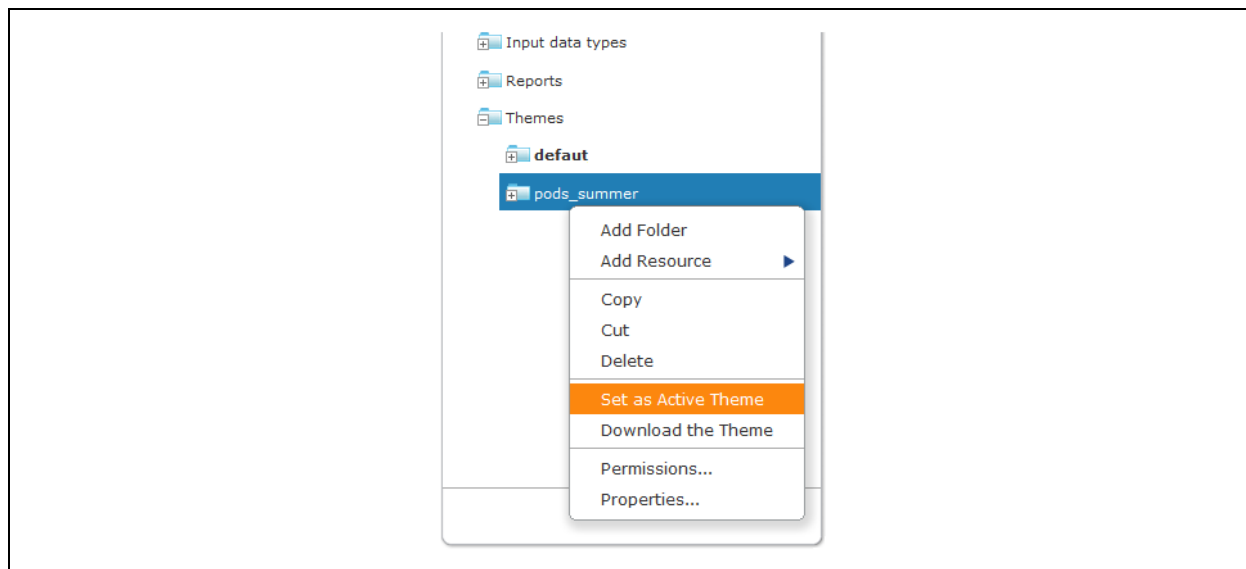
This section gives the basic procedures for administering existing themes, and for creating and modifying new theme folders. For information about how to work with CSS in themes, see section [5.4, “Working With CSS Files,” on page 90](#).

### 5.3.1 Setting the Active Theme

1. Log into JasperReports Server as administrator (`jasperadmin`).

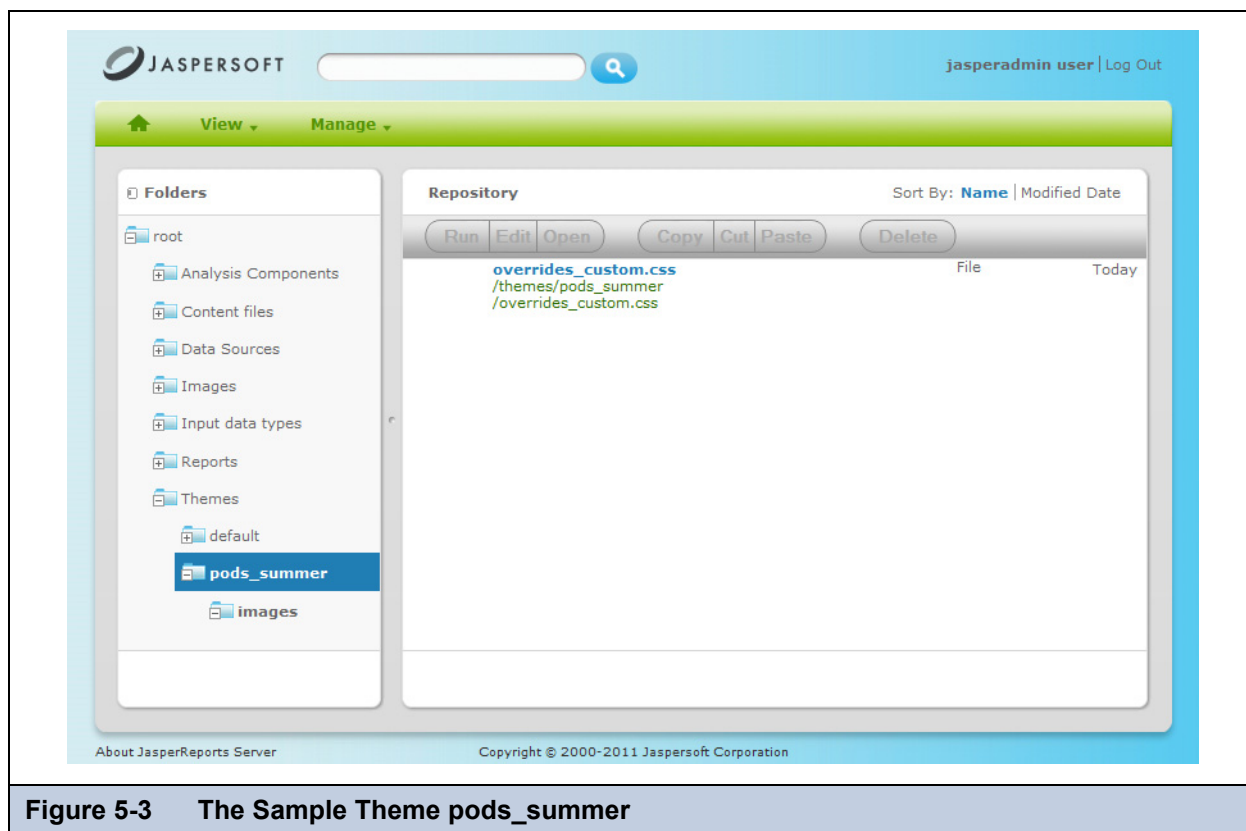
- Click **View > Repository** if necessary and expand the Themes folder.
- Right-click the new theme folder and select **Set as Active Theme**.

For example, the sample data includes a second theme called `podsummer` that you can set as active.



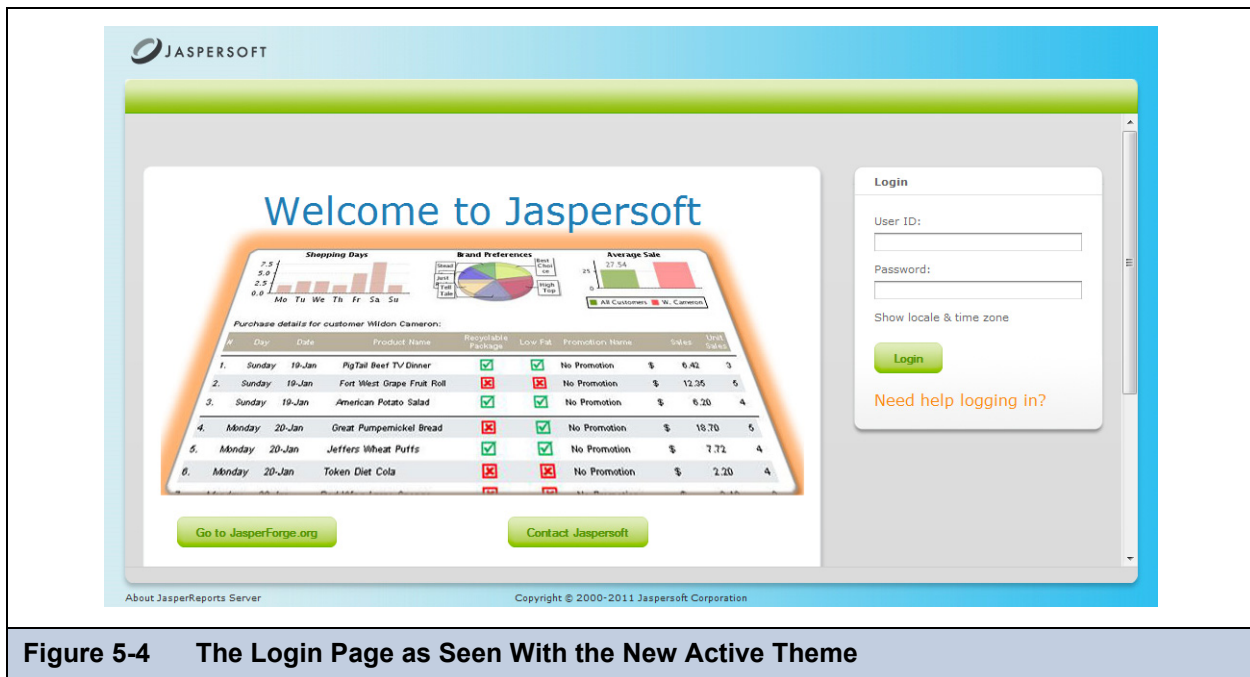
**Figure 5-2 The Set as Active Theme Context Menu Item**

As soon as the screen is refreshed, you see the effect of the new theme. Notice how the `podsummer` theme changes the colors and the logo in the user interface with just the `overrides_custom.css` file and images.



**Figure 5-3 The Sample Theme podsummer**

All users see the theme in the same way that the administrator does. Also, the theme set here applies to the login page, as shown in the following figure.



**Figure 5-4 The Login Page as Seen With the New Active Theme**

### 5.3.2 Creating Theme Folders and Files

There are two ways to create the folders and files that make up a theme:

- Create them directly as resources in the repository.
- Download and upload themes as ZIP (archive) files.

This section explains only how to store CSS files in the repository. For information about creating CSS file contents, see section 5.4, “Working With CSS Files,” on page 90.

#### 5.3.2.1 Creating Theme Folders and File Resources

A theme is simply a folder in the repository that contains CSS and image files, with optional sub-folders. Administrators can use the repository menus to create theme folders and their file resources.

##### To create theme folders and file resources:

1. Log into JasperReports Server as administrator (`jasperadmin`).
2. Click **View > Repository** if necessary and expand the Themes folder.
3. Right-click the Themes folder and select **Add Folder**. Give your folder a name and optional description as you would when creating any folder. The folder name is used as the name of the theme.



Theme folders and files can be created, copied or moved anywhere in the repository, but they can only be made active, uploaded, or downloaded when properly placed in a Themes folder.

4. Right-click your new folder and select **Add Resource > File > CSS**, and use the dialog to upload an individual CSS file. In order to be used as part of a theme, it must be one of the file names listed in section 5.2.1, “Theme Files,” on page 85.
5. To add images to your theme, create any image folders and upload image files with **Add Resource > File > Image**.
6. Repeat [step 4](#) and [step 5](#) to create all the files and images you need. If several themes use the same files or images, you can copy-paste the file resources or entire image folders from one theme to another.
7. If you need to change the contents of a CSS or image file, you can right-click it and select **Edit** to specify another file to upload and replace the current file.



If you upload CSS and image files into the active theme, the changes are visible after reloading the page in your browser.



Interacting with theme folders and files through the repository is a convenient and flexible way to create a theme. However, this method suffers from the limitation that, like other repository resources, you cannot download the files or images to edit them. For this purpose, the repository provides special download and upload actions on theme folders.

### 5.3.2.2 Downloading and Uploading Theme ZIP Files

The process of creating a theme often starts with the files of an existing theme that you modify with CSS and image editors on your computer. To support this workflow, every Themes folder has special commands for downloading and uploading themes.

Because a theme is composed of any number of files and folders, JasperReports Server uses the ZIP archive format to store a theme in a single file.

#### To download a theme ZIP file:

1. Log into JasperReports Server as administrator (`jasperadmin`).
2. Click **View > Repository** if necessary and expand the Themes folder.
3. Right-click the theme folder you want to download and select **Download the Theme**. This menu selection appears only on theme folders inside the Themes folder.
4. The server prompts you to save the file named `<theme-name>.zip`. Save it anywhere on your computer.
5. Use an archiving or compression utility to extract the files from the ZIP file and save them on your computer.

Once you have the theme files extracted on your computer, you can view the individual CSS and image files that make up the theme. For example, to create your own theme, start by downloading the default theme from the root/Themes folder. Save the extracted files on your computer and create your custom theme in another folder by copying and editing the CSS files and images of the default theme. See section 5.2.3, “CSS Priority Scheme and Custom Overrides,” on page 86 for an explanation of how to create a theme.

When you have created all the files you will need in your theme, upload it with the following procedure.

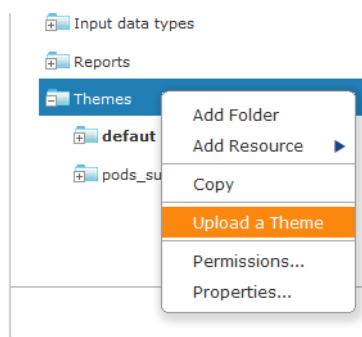
#### To upload a ZIP file as a theme:

1. Place the CSS files, optional folders, and image files that constitute your theme in a folder on your computer.
2. Use an archiving or compression utility to create a standard ZIP file of the contents of your theme folder.



The ZIP file should include only the contents of your theme, not the theme folder itself.

3. Log into JasperReports Server as administrator (`jasperadmin`).
4. Click **View > Repository** if necessary and expand the Themes folder.
5. Right-click the Themes folder and select **Upload a Theme**.



**Figure 5-5 The Upload a Theme Context Menu Item**

- In the dialog that appears, enter a name for your theme, which will be the name of its folder, and browse to find the ZIP file on your computer. Click **Upload**.



You cannot use the ZIP upload dialog to overwrite an existing theme. You must specify a theme name that doesn't already exist in the chosen Themes folder.

The server uploads your ZIP file and extracts its contents. Then it creates a folder for the new theme and creates file resources in the folder for each of the CSS and images in your ZIP file. If you had sub-folders in your theme, they are created as well.

After uploading your theme ZIP file, you can make it active to see effect of your theme on the user interface.

Creating a theme is an interactive process where you often need to make changes until you have the look and feel you want. After an initial upload, it is much easier to update individual files as described in section [5.3.2.1, “Creating Theme Folders and File Resources,” on page 88](#) than to create the ZIP file and upload it again.

## 5.4 Working With CSS Files

This section is not a CSS tutorial but rather a collection of tips and tricks for working with the CSS that makes up the themes in JasperReports server. This section focuses on how to test the themes you develop and match the CSS to its behavior in the JasperReports Server UI. Additionally, there are many different editors for CSS and tools for testing it, so the recommendations in this section are just one way of developing a theme.

### 5.4.1 Theme Development Workflow

The major choice to make when developing a theme is whether to use simple theme overrides or to duplicate and modify theme files, as described in section [5.2.3, “CSS Priority Scheme and Custom Overrides,” on page 86](#). Usually, the extent of your modifications determines which method to use.

Once you have made that determination, you are ready to create your theme. The principal steps in a theme development workflow are as follows:

| Step                                                                                                               | Reference                                                                                                                     |
|--------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------|
| 1. Download the default theme so you have a copy of the files and CSS rules that you want to modify.               | Section <a href="#">5.3.2.2, “Downloading and Uploading Theme ZIP Files,” on page 89</a> .                                    |
| 2. Create your new CSS rules, CSS files, and image files.                                                          | Section <a href="#">5.4.2, “Firebug Plug-in for FireFox,” on page 90</a> .                                                    |
| 3. Upload your new files to a test platform, and activate the theme or place them in an active theme.              | Section <a href="#">5.3.2.1, “Creating Theme Folders and File Resources,” on page 88</a> .                                    |
| 4. Verify your changes wherever they occur in the UI.                                                              | Sections <a href="#">5.4.3, “Test Platform,” on page 91</a> and <a href="#">5.4.4, “User Interface Samples,” on page 91</a> . |
| 5. Repeat <a href="#">step 2</a> through <a href="#">step 4</a> for all your changes until the theme is finalized. |                                                                                                                               |
| 6. Deploy your theme to your users.                                                                                | Section <a href="#">5.3.1, “Setting the Active Theme,” on page 86</a> .                                                       |

### 5.4.2 Firebug Plug-in for FireFox

One tool to help you find, modify, and view CSS rules in [step 2](#) above is the Firebug plug-in for the Mozilla Firefox browser. Firebug displays the HTML, JavaScript, and CSS rules of web pages as you browse. It has a dynamic interface that lets you select an element on the web page, and it displays the specific CSS rules that apply to the element. It also allows you modify those rules and immediately see the effect on the web page.

The Firebug tool is ideal for modifying themes in JasperReports Server. Once you locate the pages and elements that you want to modify, you can prototype your changes directly within the tool. For example, you can see overall effect of changing a color or modifying the spacing.

If you are implementing your theme through custom overrides, you can copy the CSS rules from the Firebug window directly into the `overrides_custom.css` file. Firebug displays the entire rule from its original file, so the copy will override it exactly. If you are modifying other files from the default theme, Firebug show you the filename and line number of the rule, so that you can easily find it in your copy of the file.

And when you are testing a theme that uses overrides, Firebug displays both the active CSS rule from `overrides_custom.css` and the original rule in the regular theme file of the inherited theme. The original rule is displayed in strike-through, so you can easily tell which rule is active and which rule it overrides.

For more information and downloads, see the [Firebug website](#).

### 5.4.3 Test Platform

When you upload a theme and make it active, it is immediately visible to every user on the server. Even editing or uploading a file into an active theme is reflected immediately in the user interface. Because developing a theme requires many iterations of uploading, activating, and testing CSS rules, you shouldn't develop themes on a production server.

In the simplest case, you can develop and test your themes before putting your JasperReports Server into production. As you test your server during the deployment, you can develop your themes without impacting real users. For deployments that are in production, you can test your themes on a second installation of the server. For example, you could install a local copy on the same computer where you develop the theme.

When your theme is well-tested and nearly complete, you can test it on the production server. Upload your theme to the Themes folder, but do not activate it. Log in as a test user and add the following parameter to the URL:

```
&theme=<theme-name>
```

This will activate your theme for the test user on all pages that you access until the user session times out. This allows you to navigate the entire application and see the effect of your theme in the production environment, without affecting other users.

On all of these test platforms, you should look at the user interface generated by your theme with the same browsers and browser versions that your users have. If you see errors, you can also use Firebug to look at the CSS rules that are involved, even if the errors do not show up on Firefox.

### 5.4.4 User Interface Samples

When testing your theme, you should look at its effect across all pages and dialogs of JasperReports Server. Your test user should access all the features of the server to view the user interface under all conditions. An additional test is to look at the user interface samples with the theme you are developing.

The user interface sample page is a new page included with the redesign of the interface. It is only accessible to administrators:

1. Log into your test environment as administrator (`jasperadmin`).
2. If you haven't already done so, upload your theme to the Themes folder and make it active.
3. Select **View > Samples** from the main menu on any page.
4. Look at all the sample components in each of the sample galleries. For example, the buttons gallery shows all the different types of buttons in every possible action state.



**Figure 5-6** Samples Gallery Showing Button Components

- When you click on the standard layouts, the sample replaces the samples page. Select **View > Samples** from the main menu again to return to the galleries.



The Samples page relies on an extra CSS file that is not required in a theme, but that can be included. The file `samples.css` is located in the default theme folder. If the sample elements do not appear as you expect, add this file to your theme and customize its rules as necessary. The rules in this file are not used anywhere else in the user interface, so it should not be included in your final theme.

Viewing the sample galleries can help you quickly find errors in your theme, especially if you are changing many rules and replacing entire CSS files. Using these samples along with the testing procedures and tools described previously, you can verify that your theme properly implements the custom user interface that you intend. Having a well-tested theme minimizes the chances of errors when you activate the theme in your production server.

## CHAPTER 6 ACCESSING REPORTS FROM IREPORT

---

iReport Designer includes the JasperReports Server plug-in. Using the JasperReports Server Plug-in, you move reports between iReport and JasperReports Server. The server plug-in uses web services to interact with the server. Some tasks you can perform using the server plug-in are:

- Browse the repository on the server from iReport.
- Add reports and subreports to the repository from iReport.
- Drag and drop images and other resources from the repository to the iReport Designer tab.
- Add and delete folders and resources on the server from iReport. Resources include images, fonts, JRXML files, JAR files, datatypes, and input controls.
- Modify resource properties on the server from iReport.
- Link input controls to reports on the server.
- Import and export data sources (JDBC, JNDI, and JavaBean).
- Download, edit, and upload JRXML files.
- Connect to multiple servers to facilitate access both test and production environments.

This chapter contains the following sections:

- **Plug-in Requirements**
- **Connecting to the Server from iReport**
- **Creating a Report Unit in the Repository from iReport**
- **Managing Resources in the Repository from iReport**
- **Editing a Report in the Repository from iReport**
- **Running a Report from iReport**

### 6.1 Plug-in Requirements


The JasperReports Server plug-in requires:

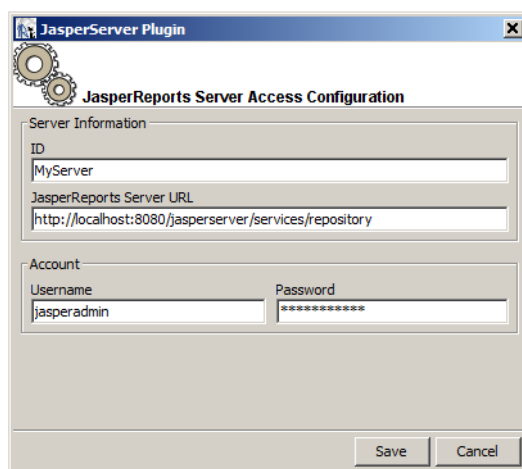
- Java Sun JDK 6 or higher
- iReport Designer 4.0 or higher – You can install iReport when you install the server.
- JasperReports Server 4.0 or higher with enabled web services – Web services are enabled by default.

## 6.2 Connecting to the Server from iReport

This procedure shows you how to connect to JasperReports Server directly from iReport.

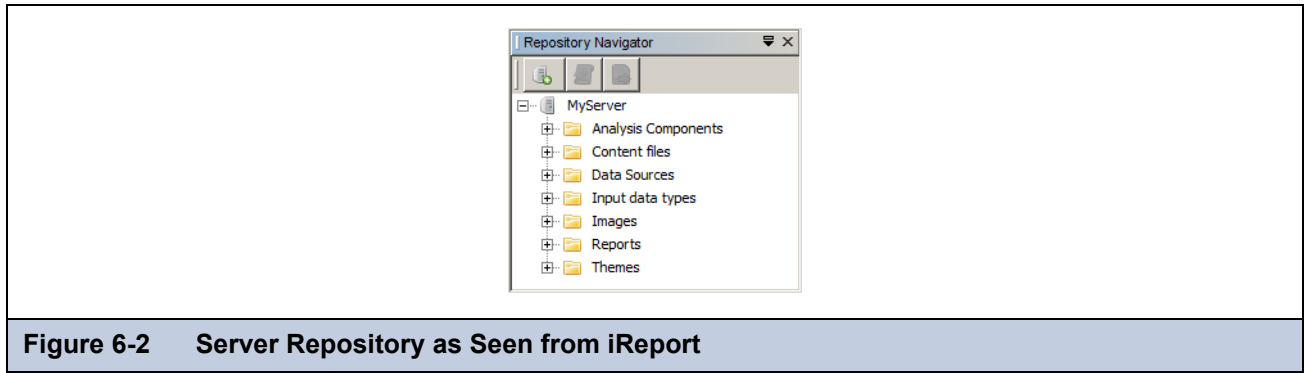
### To access the server repository within iReport:

1. Start iReport:
  - In Windows, click **Start > All Programs > JasperReports Server CP 4.1 > Start iReport Designer**.
  - In Linux, change to the iReport home directory and enter `./iReport.sh` at the command prompt.iReport opens.
2. Click **Window > JasperReports Server Repository**.  
The Repository Navigator tool bar appears.
3. On initial startup of iReport, no servers are configured. To add a server, click  in the repository navigator tool bar.  
The JasperReports Server Access Configuration dialog appears.
4. Enter the following information to access your server instance. All fields are required:
  - Name – An identifier for this server instance that will appear in the Repository Navigator.
  - JasperReports Server URL – Full URL to the repository web service; the default URL is provided. You need to change `hostname` to the name of your server:  
`http://hostname:8080/jasperserver/services/repository`
  - Username – ID of the user accessing the server from iReport.
  - Password – The password of the user.



**Figure 6-1 Configuration to Access the Server from iReport**

- Click **Save**, then expand the repository tree in the **Repository Navigator**. The repository folders appear.



**Figure 6-2 Server Repository as Seen from iReport**



The visibility of and access to resources in the Repository Navigator depend on the permissions of the user.

You can add multiple servers to the Repository Navigator. Servers configured for iReport 3.1 versions of the plug-in and later are compatible with the iReport 4.0.0 or later version of the plug-in. You may experience problems if you connect to incompatible servers.

## 6.3 Creating a Report Unit in the Repository from iReport

To add a new report unit, you need a JRXML file, which you can create in iReport beforehand. A report unit has a main JRXML file, a data source, input controls (zero or more), and resources (zero or more).

### To create a report unit in the repository:

- Connect to the server repository.
- Right-click a parent folder for the new report and click **Add > JasperServer Report**. For example, add the report unit to the Reports folder.

The ReportUnit Wizard appears.

- In Step 1, Naming, enter an ID, for example, `report_unit_example`, in the ID field. No spaces allowed.
- In the **Name** field, enter a meaningful name. For example, enter `Report Unit Example`. This name will appear in the repository on the server, and consequently, in the Repository Navigator.
- In the **Description** field, briefly describe the report unit. The description will appear in the repository on the server and help users find the report they need.

- Click **Next**.

- In Step 2, Main JRXML, locate the main JRXML file in the repository or on your hard drive. You can also specify a report that you have open in iReport.

For example, assume you have a report file named `report1.jrxml` on your hard drive and you open the report in iReport. Select **Locally Defined**, then click the **Get source from current opened report** button.

- Click **Next**.

- In Step 3, Locate Data Source, select one of the following options:

- ♦ From the repository
- ♦ Locally Defined
- ♦ Don't use any data source

For this example, select **Locally Defined**.

- Click the **Edit local datasource** button.

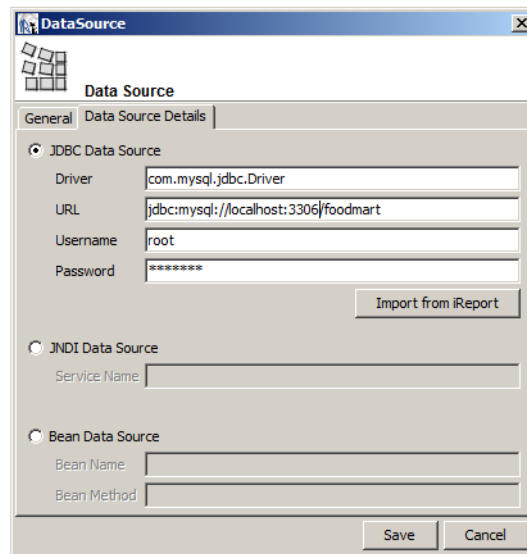
The Data Source dialog appears.

11. In the Data Source dialog:
  - a. On the **General** tab, enter a unique ID. No spaces allowed. For example, enter `MyFoodmartDS`.
  - b. In **Name**, enter a meaningful name. This name will appear in the repository on the server. For example, enter `FoodmartDS`.
  - c. On the Data Source Details tab, select a data source type, for example **JDBC Data Source**, and fill in the required details:
    - Driver – `com.mysql.jdbc.Driver`
    - URL – `jdbc:mysql://localhost:3306/foodmart`
    - User name (for the database) – `root`
    - Password (for the database)



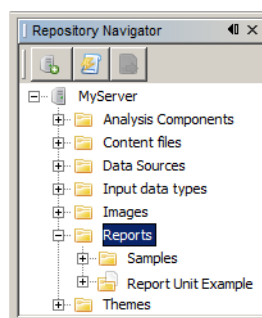
If you defined the JDBC data source in the currently opened report, you can click the **Import from iReport** button to help provide these details. Select the data source from the drop-down that you specified during report design. Fill in the port number for the database (3306 for example), and click OK

Figure 6-3 shows typical details for a JDBC data source.



**Figure 6-3 Data Source Details**

- d. Click **Save**.
12. In the ReportUnit wizard, click **Finish**.
13. In the **Repository Navigator**, expand the Reports folder. The report unit, Report Unit Example, appears.



**Figure 6-4 The New Report Unit in the Repository Navigator**





If the report unit doesn't appear in the Repository Navigator, right-click the server name (MyServer in [Figure 6-4](#)) and select **Refresh** from the context menu.

### 6.3.1 Adding an Image and Subreport to the Report Unit

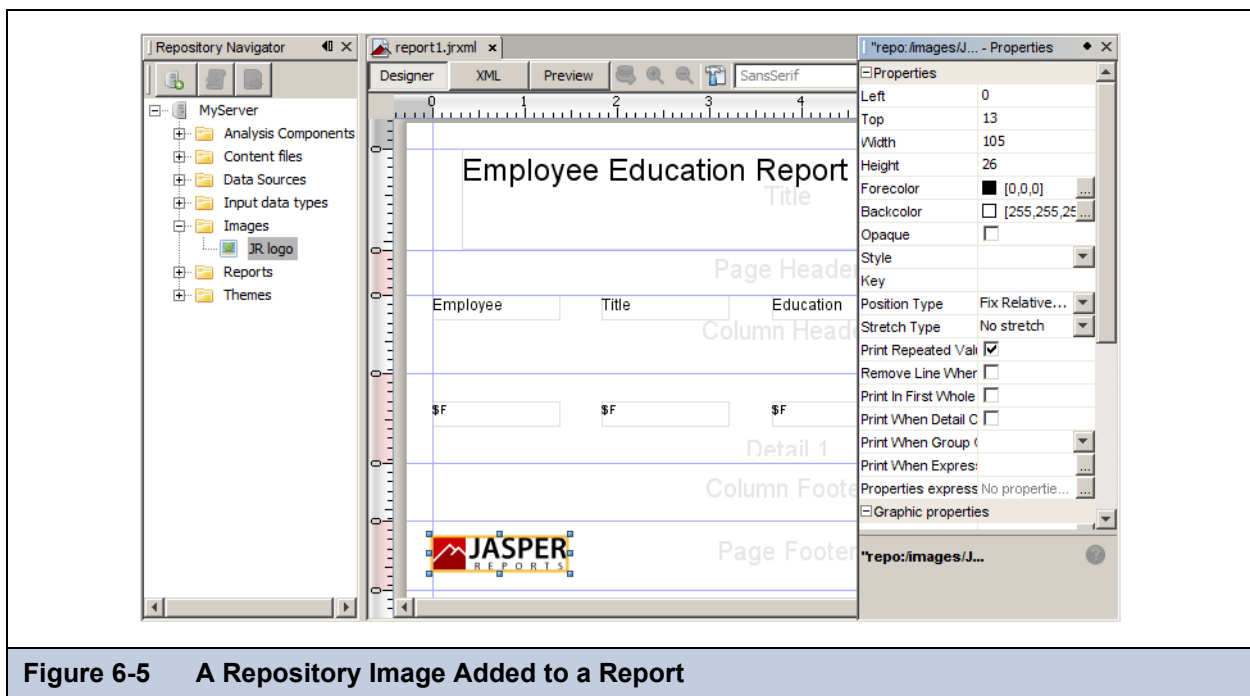
This procedure builds upon the example presented in the previous section.

#### To add an image from the repository to a report:

1. Expand the Images folder in the Repository Navigator, select an image, for example JR Logo, and drag the image to the Designer. Drop the image in the page footer band, for example.

The plug-in adds the image element to the report, setting an expression in `repo:` syntax, which refers to the image location of the image in the repository.

[Figure 6-5](#) shows the selected image file in the Repository Navigator, the image element in the page footer band of the report, and the location expression at the top and bottom of the properties pane.



**Figure 6-5 A Repository Image Added to a Report**

2. To add a subreport to a report:
  - a. From the palette, drag a **Subreport** element to the detail band of the report in the Designer. The Subreport wizard appears.
  - b. On the Subreport page of the wizard, select one of these options:
    - Create a new report
    - Use an existing report
    - Just create a subreport element

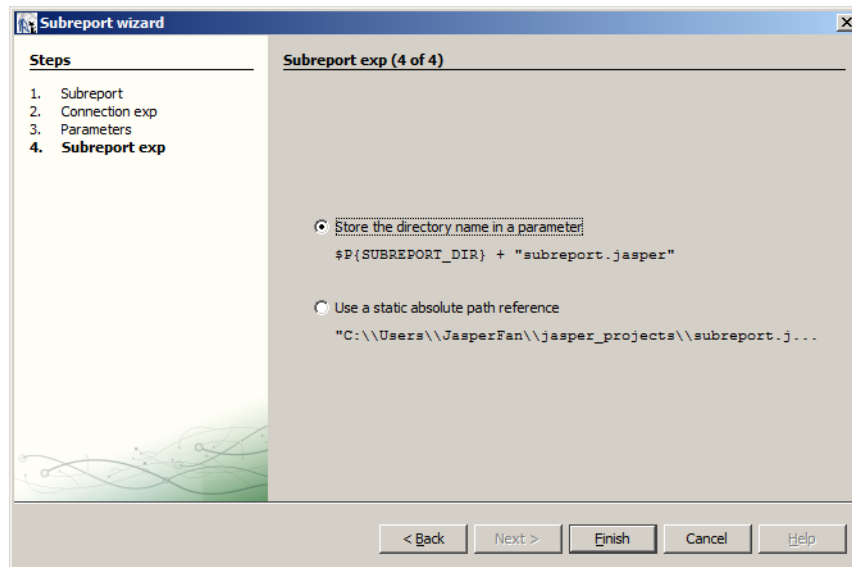
For this example, assume you have a pre-existing JRXML file that you named `subreport.jrxml`. Select **Use an existing report**.

- c. Browse to the compiled version of the report, `subreport.jasper`.
- d. Click **Next**.

The Connection expression page of the Subreport wizard appears, offering these options:

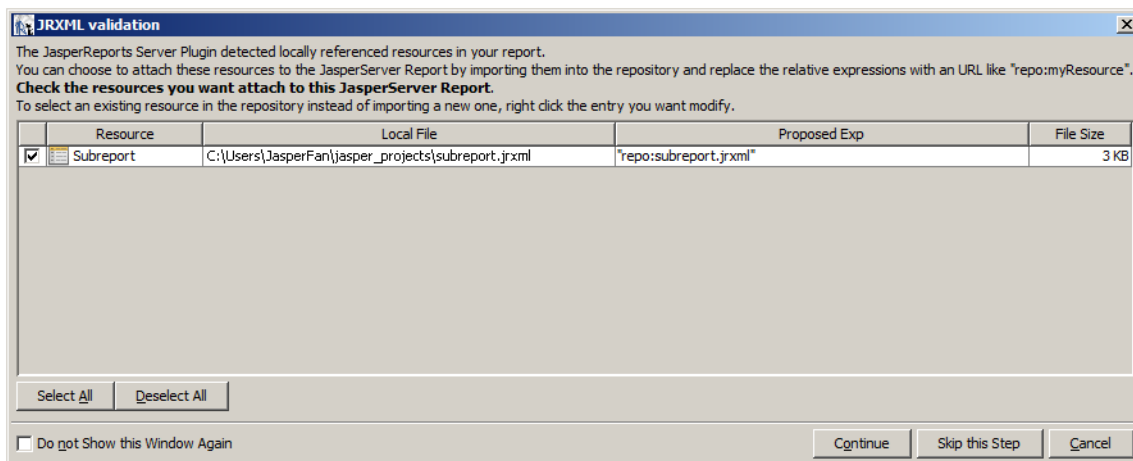
- Use the same connection used to fill the master report
- Use another connection

- ♦ Use a JRDataSource expression
  - ♦ Use an empty data source
  - ♦ Don't use any connection or data source
- e. Select the first option, **Use the same connection used to fill the master report**. Click **Next**.  
The Parameters page of the subreport wizard appears.
- f. In this example, there are no parameters. On the Parameters page, click **Next**.  
The Subreport expression page of the wizard appears, as shown in [Figure 6-6](#).
- g. On the Subreport expression page of the wizard, select **Store the directory name in a parameter**, and click **Finish**.



**Figure 6-6 A Subreport Expression**

3. In iReport, click **File > Save** to save changes to the report.
4. You must replace the Main JRXML in the repository with the one you just altered:
- a. In the Repository Navigator, expand the Report Unit Example folder.
  - b. Right-click the Main JRXML, and select **Replace with current document**.
- The JRXML validation dialog appears.



**Figure 6-7 JRXML Validation**

- c. Click Continue to upload the subreport resource and `repo:subreport.jrxml` expression to the repository.



When you upload a report to the server that uses the `repo:` syntax to refer to resources in the repository, the server prompts you to locate the resources for JRXML validation. For more information, refer to the *JasperReports Server Ultimate Guide*.

A message appears saying that the modified main JRXML successfully updated.

Click OK.

5. To run the report, right-click the report unit in the Repository Navigator and select **Run JasperServer Report** from the context menu.



You can't run a subreport alone on the server; you must run the entire report unit.

For more information, refer to section 6.6, “Running a Report from iReport,” on page 101.

## 6.4 Managing Resources in the Repository from iReport

While creating reports that you intend to store in the repository, you also need to create and manage the resources associated with them, such as images, JARs, JRXML files, property files for localized reports, input controls, datatypes, lists of values, style templates (JRTX), and data sources. If you're maintaining existing reports, you may need to access, modify, and save changes to existing resources. You can also modify the location, name, and description of the repository folders.

**To add, modify, or delete resources in the repository from iReport:**

1. If the **Repository Navigator** isn't open, click **Window > JasperReports Server Repository**.
2. To add a resource to the repository, right-click a folder, select **Add**, and then select the type of object you want to add.



If you choose to add any item other than a JasperReport, a dialog for entering the location and other information about the object appears. If you choose to add a JasperReport, a wizard guides you through the process.

3. To change the location of a resource in the repository, locate the object, then drag-and-drop it to the new location.

Or, perform the following steps:

- a. Right-click the object and select **Copy** or **Cut**. Or use the keyboard shortcuts: Ctrl-C and Ctrl-X, respectively.
- b. Locate the destination, right-click, and select **Paste**, or use the keyboard shortcut Ctrl-V.

When cutting the object, it isn't removed from its original location until pasted in the new location.

4. To delete a resource from the repository, locate the object, right-click it, and select **Delete**.

You must confirm your action to prevent accidental deletions.

5. To modify a resource, find its folder in the **Repository Navigator**, then right-click the resource and select **Properties**.

In the Properties dialog appears. The number of tabs in the dialog depends on the type of resource. The dialog for an input control has two tabs:

- ♦ General
- ♦ Input Control Details

The Properties dialog for an image has three tabs:

- ♦ General
- ♦ Resource
- ♦ Description

6. On the **General** tab, view the object's repository ID, name, and description.



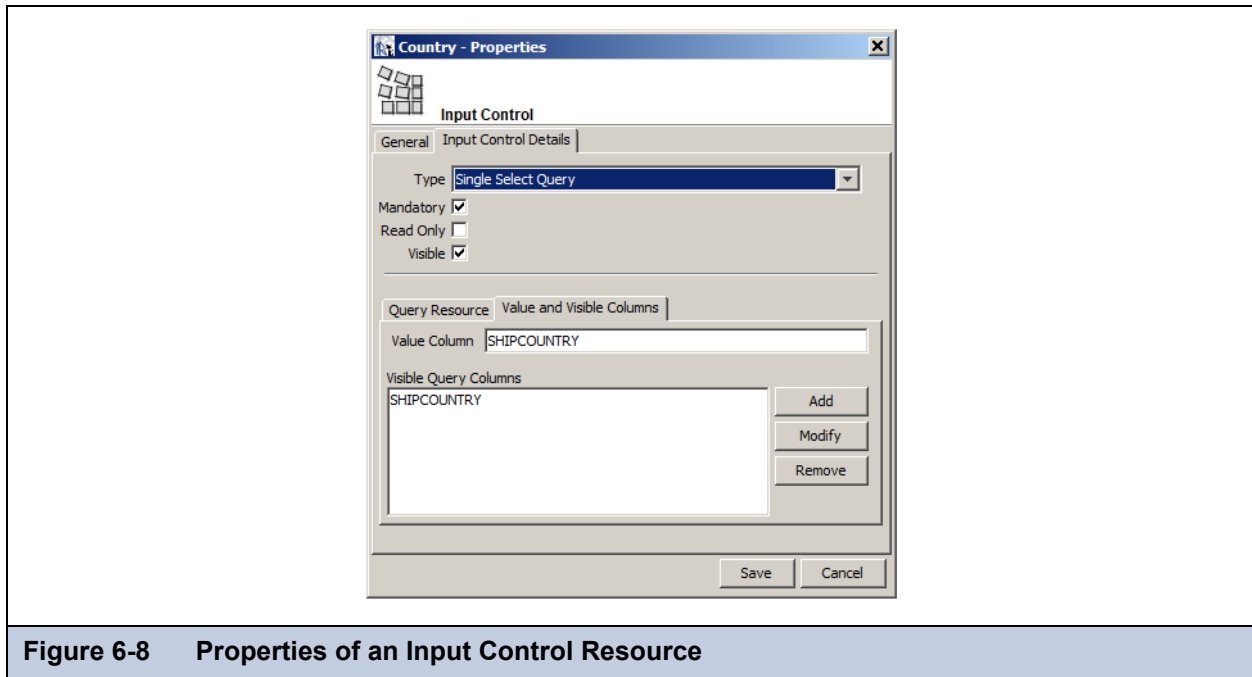
If you're logged in as an administrator, you can modify property values and save them back to the Repository.

Tabs other than **General**, if there are any, contain controls that are specific to each type of resource. File-based resources have a **Resource** tab. The image appears on the Resource tab of an image properties dialog.

7. On the **Resource** tab, check **Replace the resource with this file**, browse to another image on your hard drive, and click **Save** to replace the image on the server with the image on your hard drive.
8. Click **Export File**, and then **Save** to download a resource descriptor to your hard drive.

For example, in the Repository Navigator, you can replace a resource on one server with a resource on another. Connect to both servers using the plug-in, export an image from server 1, open image properties on server 2, and browse to the file you exported from server 1.



9. Change the design of an input control from the **Input Control Details** tab. **Figure 6-8** shows the Input Control Details properties that you can change for the Country input control.



**Figure 6-8 Properties of an Input Control Resource**

10. Click **Save** in the Properties dialog to make the changes effective in the repository.

## 6.5 Editing a Report in the Repository from iReport

In the Repository Navigator, the  icon means a report unit, and  means a JRXML file. When you work with a JRXML file in the Repository from iReport, iReport operates on a copy of the file. You need to upload the JRXML file to put it back into the repository when finished.

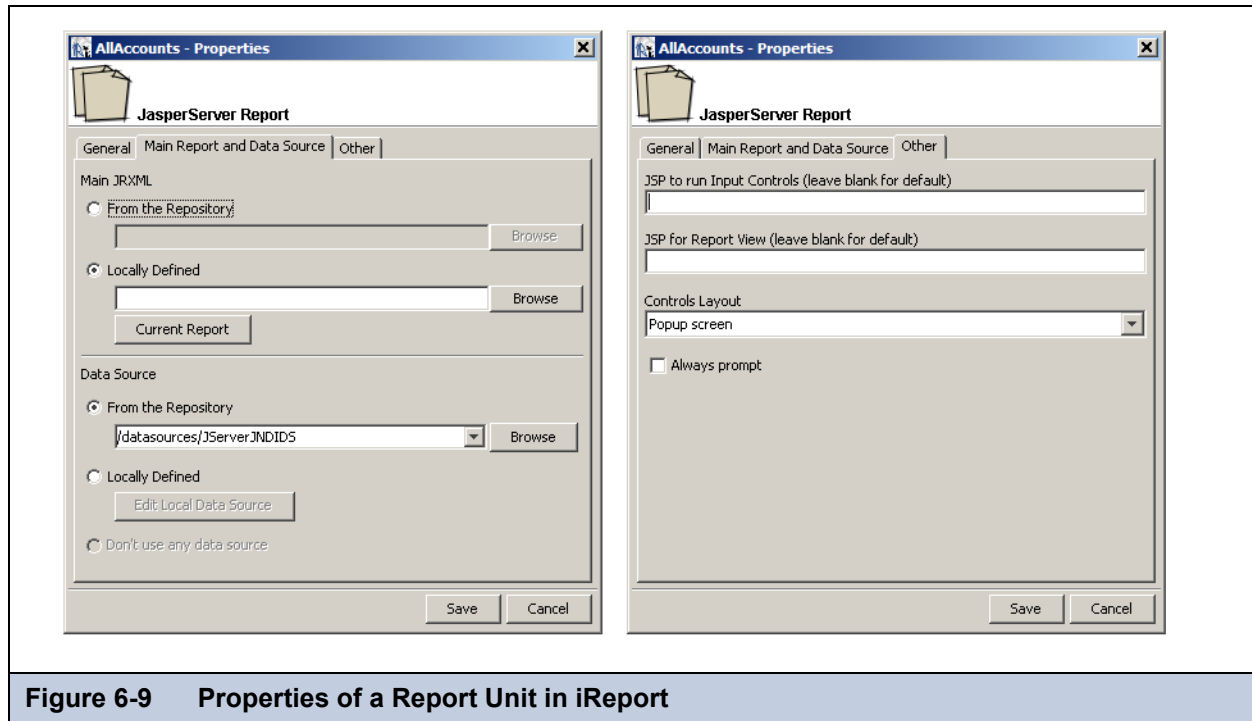
### To edit a JRXML file in the repository from iReport:

1. In the **Repository Navigator**, right-click the JRXML file in the report unit and select **Open in Editor**.  
The JRXML from the repository is stored locally in the <USER\_HOME>/.ireport/jstmp directory. This directory is never automatically cleared.  
The JRXML appears in the Designer window.
2. Edit the file as needed, either in the Designer or in the XML editor. For information about adding images or subreports to the JRXML file, see section 6.3, “**Creating a Report Unit in the Repository from iReport,**” on page 95.
3. To save the file and upload it to the repository, right-click the file in the Repository Navigator and click **Replace with Current JRXML**.  
A message confirms the operation’s success.

**To edit a report unit in the repository from iReport:**

1. In the Repository Navigator, right-click the report unit and select **Properties**.
2. On the **General** tab, change the repository name or description.
3. On the **Main Report and Data Source** tab, you can change the JRXML file for the report, either by selecting one from the repository, or uploading one through iReport. Click **Current Report** to use the JRXML currently being edited in iReport. Similarly, you can select the data source from the repository or from iReport. See [Figure 6-9](#).
4. On the **Other** tab, set display properties for any input controls.
5. In the JSP fields, modify the appearance of the controls.

The Controls Layout determines where they appear.



**Figure 6-9 Properties of a Report Unit in iReport**

6. Click **Save** to make the changes effective in the repository.

## 6.6 Running a Report from iReport

**To run a report on the server using all the export options set in iReport:**

1. Right-click a report in the Repository Navigator and click **Run JasperServer Report**.
2. If the report has input controls, the Report Parameters dialog prompts you for values.

The report is displayed in the report viewer. If the report doesn't appear, look for errors or messages in Report Problems at the bottom of the iReport console.



## GLOSSARY

---

### **Ad Hoc Editor**

The interactive report designer in JasperReports Server Professional and Enterprise editions. Starting from a collection of fields predefined in a Topic or selected from a Domain, the Ad Hoc Editor lets you drag and drop report elements to draft, preview, and finalize reports. Like JRXML reports, Ad Hoc reports can be run, printed, and scheduled within JasperReports Server. In addition, Ad Hoc reports may be reopened in the Ad Hoc Editor, further modified, and saved.

### **Audit Archiving**

To prevent audit logs from growing too large to be easily accessed, the system installer configures JasperReports Server to move current audit logs to an archive after a certain number of days, and to delete logs in the archive after a certain age. The archive is another table in the JasperReports Server's private database.

### **Audit Domains**

A Domain that accesses audit data in the repository and lets administrators create Ad Hoc reports of server activity. There is one Domain for current audit logs and one for archived logs.

### **Audit Logging**

When auditing is enabled, audit logging is the active recording of who used JasperReports Server to do what when. The system installer can configure what activities to log, the amount of detail gathered, and when to archive the data. Audit logs are stored in the same private database that JasperReports Server uses to store the repository, but the data is only accessible through the audit Domains.

### **Auditing**

A feature of JasperReports Server Enterprise edition that records all server activity and allows administrators to view the data.

### **Calculated Field**

In a Domain, a field whose value is calculated from a user-written formula that may include any number of fields, operators, and constants. A calculated field is defined in the Domain Designer, and it becomes one of the items to which the Domain's security file and locale bundles can apply.

### **CRM**

Customer Relationship Management. The practice of managing every facet of a company's interactions with its clientele. CRM applications help businesses track and support their customers.

### **CrossJoin**

An MDX function that combines two or more dimensions into a single axis (column or row).

## Cube

The basis of most OLAP applications, a cube is a data structure that contains three or more dimensions that categorize the cube's quantitative data. When you navigate the data displayed in an OLAP view, you are exploring a cube.

## Custom Field

In the Ad Hoc Editor, a field that is created through menu items as a simple function of one or two available fields, including other custom fields. When a custom field becomes too complex or needs to be used in many reports, it is best to define it as a calculated field in a Domain.

## Dashboard

A collection of reports, input controls, graphics, labels, and web content displayed in a single, integrated view. Dashboards often present a high level view of your data, but input controls can parameterize the data to display. For example, you can narrow down the data to a specific date range. Embedded web content, such as other web-based applications or maps, make dashboards more interactive and functional.

## Derived Table

In a Domain, a derived table is defined by an additional query whose result becomes another set of items available in the Domain. For example, with a JDBC data source, you can write an SQL query that includes complex functions for selecting data. You can use the items in a derived table for other operations on the Domain, such as joining tables, defining a calculated field, or filtering. The items in a derived table can also be referenced in the Domain's security file and locale bundles.

## Data Policy

In JasperReports Server, a setting that determines how the server processes and caches data used by Ad Hoc reports. Select your data policies by clicking **Manage > Ad Hoc Settings**.

## Data Source

Defines the connection properties that JasperReports Server needs to access data. The server transmits queries to data sources and obtains datasets in return for use in filling reports and previewing Ad Hoc reports. JasperReports Server supports JDBC, JNDI, and Bean data sources; custom data sources can be defined as well.

## Dataset

A collection of data arranged in columns and rows. Datasets are equivalent to relational results sets and the `JRDataSource` type in the JasperReports Library.

## Datatype

In JasperReports Server, a datatype is used to characterize a value entered through an input control. A datatype must be of type text, number, date, or date-time. It can include constraints on the value of the input, for example maximum and minimum values. As such, a datatype in JasperReports Server is more structured than a datatype in most programming languages.

## Denormalize

A process for creating table joins that speeds up data retrieval at the cost of having duplicate row values between some columns.

## Dice

An OLAP operation to select columns.

## Dimension

A categorization of the data in a cube. For example, a cube that stores data about sales figures might include dimensions such as time, product, region, and customer's industry.

## Domain

A virtual view of a data source that presents the data in business terms, allows for localization, and provides data-level security. A Domain is not a view of the database in relational terms, but it implements the same functionality within JasperReports Server. The design of a Domain specifies tables in the database, join clauses, calculated fields, display names, and default properties, all of which define items and sets of items for creating Ad Hoc reports.



**Domain Topic**

A Topic that is created from a Domain by the Data Chooser. A Domain Topic is based on the data source and items in a Domain, but it allows further filtering, user input, and selection of items. Unlike a JRXML-based Topic, a Domain Topic can be edited in JasperReports Server by users with the appropriate permissions.

**Drill**

To click on an element of an OLAP view to change the data that is displayed:

- Drill down. An OLAP operation that exposes more detailed information down the hierarchy levels by delving deeper into the hierarchy and updating the contents of the navigation table.
- Drill through. An OLAP operation that displays detailed transactional data for a given aggregate measure. Click a fact to open a new table beneath the main navigation table; the new table displays the low-level data that constitutes the data that was clicked.
- Drill up. An OLAP operation for returning the parent hierarchy level to view to summary information.

**Eclipse**

An open source Integrated Development Environment (IDE) for Java and other programming languages, such as C/C++.

**ETL**

Extract, Transform, Load. A process that retrieves data from transactional systems, and filters and aggregates the data to create a multidimensional database. Generally, ETL prepares the database that your reports will access. The Jaspersoft ETL product lets you define and schedule ETL processes.

**Fact**

The specific value or aggregate value of a measure for a particular member of a dimension. Facts are typically numeric.

**Field**

A field is equivalent to a column in the relational database model. Fields originate in the structure of the data source, but you may define calculated fields in a Domain or custom fields in the Ad Hoc Editor. Any type of field, along with its display name and default formatting properties, is called an item and may be used in the Ad Hoc Editor.

**Frame**

A dashboard element that displays reports or custom URLs. Frames can be mapped to input controls if their content can accept parameters.

**Group**

In a report, a group is a set of data rows that have an identical value in a designated field.

- In a table, the value appears in a header and footer around the rows of the group, while the other fields appear as columns.
- In a chart, the field chosen to define the group becomes the independent variable on the X axis, while the other fields of each group are used to compute the dependent value on the Y axis.

**Hierarchy Level**

In an OLAP cube, a member of a dimension containing a group of members.

**Input Control**

A button, check box, drop-down list, text field, or calendar icon that allows users to enter a value when running a report or viewing a dashboard that accepts input parameters. For JRXML reports, input controls and their associated datatypes must be defined as repository objects and explicitly associated with the report. For Domain-based reports that prompt for filter values, the input controls are defined internally. When either type of report is used in a dashboard, its input controls are available to be added as special content.

**iReport Designer**

An open source tool for graphically designing reports that leverage all features of the JasperReports Library. The Jaspersoft iReport Designer lets you drag and drop fields, charts, and sub-reports into a canvas, and also define parameters or expressions for each object to create pixel-perfect reports. iReport Designer outputs the JRXML of the report or uploads it directly to JasperReports Server.

### Item

When designing a Domain or creating a Topic based on a Domain, an item is the representation of a database field or a calculated field along with its display name and formatting properties defined in the Domain. Items can be grouped in sets and are available for use in the creation of Ad Hoc reports.

### JasperReports Library

An embeddable, open source, Java API for generating a report, filling it with current data, drawing charts and tables, and exporting to any standard format (HTML, PDF, Excel, CSV, and others). JasperReports processes reports defined in JRXML, an open XML format that allows the report to contain expressions and logic to control report output based on run-time data.

### JasperReports Server

A commercial open source, server-based application that calls the JasperReports library to generate and share reports securely. JasperReports Server authenticates users and lets them upload, run, view, schedule, and send reports from a web browser. Commercial versions provide metadata layers, interactive report and dashboard creation, and enterprise features such as organizations and auditing.

### Jaspersoft ETL

A graphical tool for designing and implementing your data extraction, transforming, and loading (ETL) tasks. It provides hundreds of data source connectors to extract data from many relational and non-relational systems. Then, it schedules and performs data aggregation and integration into data marts or data warehouses that you use for reporting.

### Jaspersoft OLAP

A relational OLAP server integrated into JasperReports Server that performs data analysis with MDX queries. The product includes query builders and visualization clients that help users explore and make sense of multidimensional data. Jaspersoft OLAP also supports XML/A connections to remote servers.

### JavaBean

A reusable Java component that can be dropped into an application container to provide standard functionality.

### JDBC

Java Database Connectivity. A standard interface that Java applications use to access databases.

### JNDI

Java Naming and Directory Interface. A standard interface that Java applications use to access naming and directory services.

### Join Tree

In Domains, a collection of joined tables from the actual data source. A join is the relational operation that associates the rows of one table with the rows of another table based on a common value in given field of each table. Only the fields in a same join tree or calculated from the fields in a same join tree may appear together in a report.

### JPivot

An open source graphical user interface for OLAP operations. For more information, visit <http://jpivot.sourceforge.net/>.

### JRXML

An XML file format for saving and sharing reports created for the JasperReports Library and the applications that use it, such as iReport Designer and JasperReports Server. JRXML is an open format that uses the XML standard to define precisely all the structure and configuration of a report.

### MDX

Multidimensional Expression Language. A language for querying multidimensional objects, such as OLAP (On Line Analytical Processing) cubes, and returning cube data for analytical processing. An MDX query is the query that determines the data displayed in an OLAP view.

### Measure

Depending on the context:

- In a report, a formula that calculates the values displayed in a table's columns, a crosstab's data values, or a chart's dependent variable (such as the slices in a pie).
- In an OLAP view, a formula that calculates the facts that constitute the quantitative data in a cube.

**Mondrian**

A Java-based, open source multidimensional database application.

**Mondrian Connection**

An OLAP client connection that consists of an OLAP schema and a data source used to populate an OLAP view.

**Mondrian Schema Editor**

An open source Eclipse plug-in for creating Mondrian OLAP schemas.

**Mondrian XMLA Source**

A server-side XMLA source definition of a remote client-side XML/A connection used to populate an OLAP view using the XMLA standard.

**MySQL**

An open source relational database management system. For information, visit <http://www.mysql.com/>.

**Navigation Table**

The main table in an OLAP view that displays measures and dimensions as columns and rows.

**ODBO Connect**

Jaspersoft ODBO Connect enables Microsoft Excel 2003 and 2007 Pivot Tables to work with Jaspersoft OLAP and other OLAP servers that support the XML/A protocol. After setting up the Jaspersoft ODBO data source, business analysts can use Excel Pivot Tables as a front-end for OLAP analysis.

**OLAP**

On Line Analytical Processing. Provides multidimensional views of data that help users analyze current and past performance and model future scenarios.

**OLAP Client Connection**

A definition for retrieving an OLAP view. An OLAP client connection is either a direct Java connection (Mondrian connection) or an XML-based API connection (XML/A connection).

**OLAP Schema**

A metadata definition of a multidimensional database. In Jaspersoft OLAP, schemas are stored in the repository as XML file resources.

**OLAP View**

Also called an analysis view. A view of multidimensional data that is based on an OLAP client connection and an MDX query. It is the entry point to analysis operations, such as slice and dice, drill down, and drill through.

**Organization**

A set of users that share folders and resources in the repository. An organization has its own user accounts, roles, and root folder in the repository to securely isolate it from other organizations that may be hosted on the same instance of JasperReports Server.

**Organization Admin**

Also called the organization administrator. A user in an organization with the privileges to manage the organization's user accounts and roles, repository permissions, and repository content. An organization admin can also create sub-organizations and manage all of their accounts, roles, and repository objects. The default organization admin in each organization is the `jasperadmin` account.


## Outlier

A fact that seems incongruous when compared to other member's facts. For example, a very low sales figure or a very high number of helpdesk tickets. Such outliers may indicate a problem (or an important achievement) in your business. The analysis features of Jaspersoft OLAP excel at revealing outliers.

## Parameter

Named values that are passed to the engine at report-filling time to control the data returned or the appearance and formatting of the report. A report parameter is defined by its name and type. In JasperReports Server, parameters can be mapped to input controls that users can interact with.

## Pivot

To rotate a crosstab such that its row groups become column groups and its column groups become rows. In the Ad Hoc Editor, pivot a crosstab by clicking .

## Pivot Table

A table with two physical dimensions (for example, X and Y axis) for organizing information containing more than two logical dimensions (for example, PRODUCT, CUSTOMER, TIME, and LOCATION), such that each physical dimension is capable of representing one or more logical dimensions, where the values described by the dimensions are aggregated using a function such as SUM. Pivot tables are used in Jaspersoft OLAP.

## Properties

Settings associated with an object. The settings determine certain features of the object, such as its color and label. Properties are normally editable. In Java, properties can be set in files listing objects and their settings.

## Repository

The tree structure of folders that contain all saved reports, dashboards, OLAP views, and resources. Users access the repository through the JasperReports Server web interface or through iReport. Applications can access the repository through the web service API. Administrators use the import and export utilities to back up the repository contents.

## Resource

In JasperReports Server, anything residing in the repository, such as an image, file, font, data source, Topic, Domain, report element, saved report, report output, dashboard, or OLAP view. Resources also include the folders in the repository. Administrators set user and role-based access permissions on repository resources to establish a security policy.

## Role

A security feature of JasperReports Server. Administrators create named roles, assign them to user accounts, and then set access permissions to repository objects based on those roles. Certain roles also determine what functionality and menu options are displayed to users in the JasperReports Server interface.

## Schema

A logical model that determines how data is stored. For example, the schema in a relational database is a description of the relationships between tables, views, and indexes. In Jaspersoft OLAP, an OLAP schema is the logical model of the data that appears in an OLAP view; they are uploaded to the repository as resources. For Domains, schemas are represented in XML design files.

## Schema Workbench

A graphical tool for easily designing OLAP schemas, data security schemas, and MDX queries. The resulting cube and query definitions can then be used in Jaspersoft OLAP to perform simple but powerful analysis of large quantities of multi-dimensional data stored in standard RDBMS systems.

## Set

In Domains and Domain Topics, a named collection of items grouped together for ease of use in the Ad Hoc Editor. A set can be based on the fields in a table or entirely defined by the Domain creator, but all items in a set must originate in the same join tree. The order of items in a set is preserved.

**Slice**

An OLAP operation for filtering data rows.

**SQL**

Structured Query Language. A standard language used to access and manipulate data and schemas in a relational database.

**System Admin**

Also called the system administrator. A user who has unlimited access to manage all organizations, users, roles, repository permissions, and repository objects across the entire JasperReports Server instance. The system admin can create root-level organizations and manage all server settings. The default system admin is the `superuser` account.

**Topic**

A JRXML file created externally and uploaded to JasperReports Server as a basis for Ad Hoc reports. Topics are created by business analysts to specify a data source and a list of fields with which business users can create reports in the Ad Hoc Editor. Topics are stored in the Ad Hoc Components folder of the repository and displayed when a user launches the Ad Hoc Editor.

**Transactional Data**

Data that describe measurable aspects of an event, such as a retail transaction, relevant to your business. Transactional data are often stored in relational databases, with one row for each event and a table column or field for each measure.

**User**

Depending on the context:

- A person who interacts with JasperReports Server through the web interface. There are generally three categories of users: administrators who install and configure JasperReports Server, database experts or business analysts who create data sources and Domains, and business users who create and view reports and dashboards.
- A user account that has an ID and password to enforce authentication. Both people and API calls accessing the server must provide the ID and password of a valid user account. Roles are assigned to user accounts to determine access to objects in the repository.

**WCF**

Web Component Framework. A low-level GUI component of JPivot. For more information, see <http://jpivot.sourceforge.net/wcf/index.html>.

**Web Services**

A SOAP (Simple Object Access Protocol) API that enables applications to access certain features of JasperReports Server. The features include repository, scheduling and user administration tasks.

**XML**

eXtensible Markup language. A standard for defining, transferring, and interpreting data for use across any number of XML-enabled applications.

**XML/A**

XML for Analysis. An XML standard that uses Simple Object Access protocol (SOAP) to access remote data sources. For more information, see <http://www.xmla.org/>

**XML/A Connection**

A type of OLAP client connection that consists of Simple Object Access Protocol (SOAP) definitions used to populate an OLAP view.



## INDEX

### Symbols

&theme parameter 91

### A

access control

- administering 76
- authentication 75
- authorization 75
- framework for 75
- object-level permissions 78
- permissions 76

access grants, scheduled reports 19

Add Resource 30

adding reports 38

administering JasperReports Server

- access control 76
- logging in as a user 80
- logging in as an administrator 68
- Manage menu 67
- object-level permissions 78
- passwords 80
- roles 71
- users 67, 75

Always prompt option 52, 55

authentication. *See* access control.

authorization. *See* access control.

### B

browsing the repository 9

### C

cascading input controls 18, 54

controls. *See* input controls.

creating reports 38

CSS files 85, 90

### D

data sources, access grants, and scheduled reports 19

datatypes 42, 45

defaults

input controls 42

roles 71

users 67

### E

Event Details page 28

external resources 40

### F

Firebug plug-in 90

folders 99

formats for report output 16

framework for access control 75

### I

input controls

- adding 42
- Always prompt option 52
- cascading 18, 54
- check box type 46
- date type 48
- default values 16, 42, 52
- drop-down type 47
- list type 47
- Optional JSP Location option 52
- pop-up page 17
- query type 49
- running a report with input controls 16
- text 44
- types 42
- Visible property 42, 52

installing JasperReports Server plug-in 93

iReport

- adding resources 99
- creating reports 95
- JRXML files 100
- repository 95, 100
- running reports 101

**J**

- JAR files 38, 40
- JasperReport wizard
  - Controls & Resources 31–34, 39–52, 55
  - Data Source 34, 52
  - edit report unit 55
  - Query 34–37
  - Set Up 30–31, 39
- JasperReports samples 14
- Jaspersoft OLAP 9
- Jobs List page 23
- jobs. *See* scheduling reports.
- JRXML files
  - external resources 40
  - field names in 56
  - in iReport 100
  - to create a new report 31, 38

**L**

- locales 8, 56
- Locate File Resource page 33, 34
- Locate Query page 35
- logging in
  - administering 80
  - as a user 80
  - as administrator 68
  - procedure 8

**M**

- Manage menu, overview 9
- messages about system events 27
- Messages page 28

**O**

- OLAP views 76
- overrides\_custom.css file 86

**P**

- pages
  - Event Details 28
  - Input Control 17
  - Jobs List 23
  - Locate File Resource 33, 34
  - Locate Query 35
  - Messages 28
  - Report Viewer 17
  - Set Up the Job 20
  - Set Up the Report 30, 31, 39
- parametrized queries
  - adding input controls 42
  - and reports 26
  - running a report with 16
- passwords
  - expiration 80
  - users changing 80, 81
- permissions. *See* access control.
- plug-ins, JasperReports Server 93
- profile attributes 68, 76

**Q**

- query resource 35

**R**

- recurrence
  - calendar recurrence 25
  - types 20
- Report Viewer 17
- reports
  - See also* scheduling reports *and* running reports.
  - adding 38
  - creating 38, 95
  - effect of access grants 19
  - exporting report output 16
  - localization 56
  - report output 16
  - report parameters 26
  - running 15, 20, 26, 101
  - saving 52
  - saving report output 16
  - submitting 37
  - with input controls 16
- repository
  - in iReport 95, 100
  - list of reports 23
  - overview 9
  - searching 8, 73
  - system messages 27
- repository page 9
- resources
  - adding to iReport 99
  - and localizations 56
- roles
  - administering 71
  - and users 71
  - default roles 71
  - object-level permissions 78
  - searching for 73
- running reports
  - See also* reports *and* scheduling reports.
  - example 15, 16
  - from iReport 101
  - in the background 26
  - on a schedule 20

**S**

- sample files
  - AllAccounts.jrxml 30, 31
  - for JasperReports 14
  - logo.jpg 30
  - SalesByMonth.jrxml 39
  - SalesByMonthDetail.jrxml 39
- sample reports
  - Accounts 16
  - Freight 17
  - SalesByMonth 39



- saving reports 52
  - ScheduledTime parameter 26
  - Scheduler wizard 20–22
  - scheduling reports
    - See also* reports *and* running reports.
    - access grants and 19
    - defining a job 20
    - defining schedules 19
    - editing a job 23
    - recurrence 24
    - viewing job schedules 23
  - search field 10
  - searching the repository 73
  - Set Up the Job page 20
  - Set Up the Report page 30, 31, 39
  - submitting reports 37
- T**
- themes
    - &theme parameter 91
    - CSS files 85, 90
    - custom overrides 86
    - downloading 89
    - Firebug plug-in 90
    - inheritance mechanism 85
    - sample galleries 91
    - Themes folder 84
    - uploading 89
  - URL parameter 91
  - ZIP archive 88
  - Themes folder 84
  - time zone 8
- U**
- user interface samples 91
  - users
    - administering 67, 75
    - authenticating users 75, 76
    - authorizing users 75
    - changing passwords 80, 81
    - default users 67
    - logging in as a user 80
    - object-level permissions 78
    - roles 71
  - utilities
    - iReport 93
    - messages 27
- V**
- View menu, overview 9
  - viewing scheduled reports 19
  - views 76
- W**
- wizards
    - JasperReport. *See* JasperReport wizard.
    - Scheduler 20–22

