# Pentaho BI Suite Administrator's Guide

# Help and Support Resources

If you have questions that are not covered in this guide, or if you would like to report errors in the documentation, please contact your Pentaho technical support representative.

Support-related questions should be submitted through the Pentaho Customer Support Portal at http://support.pentaho.com.

For information about how to purchase support or enable an additional named support contact, please contact your sales representative, or send an email to sales@pentaho.com.

For information about instructor-led training on the topics covered in this guide, visit http://www.pentaho.com/training.

# Limits of Liability and Disclaimer of Warranty

The author(s) of this document have used their best efforts in preparing the content and the programs contained in it. These efforts include the development, research, and testing of the theories and programs to determine their effectiveness. The author and publisher make no warranty of any kind, express or implied, with regard to these programs or the documentation contained in this book.

The author(s) and Pentaho shall not be liable in the event of incidental or consequential damages in connection with, or arising out of, the furnishing, performance, or use of the programs, associated instructions, and/or claims.

# Trademarks

Pentaho (TM) and the Pentaho logo are registered trademarks of Pentaho Corporation. All other trademarks are the property of their respective owners. Trademarked names may appear throughout this document. Rather than list the names and entities that own the trademarks or insert a trademark symbol with each mention of the trademarked name, Pentaho states that it is using the names for editorial purposes only and to the benefit of the trademark owner, with no intention of infringing upon that trademark.

# Company Information

Pentaho Corporation
Citadel International, Suite 340
5950 Hazeltine National Drive
Orlando, FL 32822
Phone: +1 407 812-OPEN (6736)
Fax: +1 407 517-4575
http://www.pentaho.com

E-mail: communityconnection@pentaho.com

Sales Inquiries: sales@pentaho.com

Documentation Suggestions: documentation@pentaho.com

Sign-up for our newsletter: http://community.pentaho.com/newsletter/

# Contents

# Introduction

This administrator's guide provides you with instructions for planning, deploying, and administering the Pentaho BI Platform. The majority of the instructions in this guide involve using the Pentaho Enterprise Console (a standalone utility with a thin client interface), but there is also an appendix containing command line equivalents to all possible BI Platform configuration tasks.

## Prerequisites

Console features and functions are completely dependent on the BI Server. You must have a BI Server up and running prior to starting your Pentaho Enterprise Console.

### Identifying File Paths

Most sample file paths in this document are based on the installation wizard. If you performed a manual installation, the file paths may be slightly different. In most scenarios, the Pentaho Enterprise Console is in `/pentaho/server/enterprise-console/`.

## Is This Guide For You?

This document is written for system administrators, DBAs, IT managers, or others who are responsible for the day-to-day tasks associated with administering a Pentaho deployment. It is assumed that you, the reader, have working knowledge of your Pentaho instance and all of the services and technologies that connect to and work with it.

# Starting and Stopping the Server and Pentaho Enterprise Console

This section provides you with instructions for starting and stopping the server and console depending on the type of operating system you are using.

## Starting the BI Server in Windows

To start the server in Windows...

1. Navigate to `\pentaho\server\biserver-ee\`.
2. Double-click to run the **start-pentaho.bat** file.
   The batch file runs and the launches the Tomcat application server.
3. Open a browser and navigate to `http://localhost:8080/` (modify this hostname and port number to match your BI Server configuration).

The BI Server is now started. To start it automatically as a service at boot time, consult the installation guide you used when installing the BI Server.

## Starting the BI Server in Linux

To start the BI server in Linux...

1. Open a terminal and navigate to `.../pentaho/server/biserver-ee`.
2. Start the BI Server using the following command, **./start-pentaho.sh**.
3. Open a browser and navigate to `http://localhost:18080/` or `http:yourhostname:8080`.

## Starting the BI Server in OS X

To start the server in Mac OS X...

1. Open a command window.
2. Navigate to the `.../pentaho/server/biserver-ee` directory.
3. Type **./start-pentaho.sh** and press **RETURN**.
   Once the server is running, the following line appears in the command window: **Console is started. Console can be accessed using http://example.com:8080 or http://127.0.0.1:8080**.

## Stopping the BI Server

To stop the server...

1. Open a command window.
2. Go to the install directory that contains the file, **stop-pentaho.bat** (or **.sh**).
3. Type **stop-pentaho.bat** (or **.sh** and press **ENTER** (or **RETURN**) to stop the server.

## Starting the Pentaho Enterprise Console in Windows

Make sure your server is running before you start the console.

To start the console...

1. Open a command window.
2. Run the `...\pentaho\server\enterprise-console\start-pec.bat` script.

Once the console is running, the following line appears in the command window: **Console is started. Console can be accessed using http://example.com:8088 or http://127.0.0.1:8088**, replacing the example hostname, IP address, and port number with that of the machine running the Pentaho Enterprise Console.

# Starting the Pentaho Enterprise Console in Linux and Solaris

To start the console in Linux...

1. Open the command window
2. Run the `.../pentaho/server/enterprise-console/start-pec.sh` script.
   Once the console is running, the following line appears in the command window: **Console is started. Console can be accessed using http://example.com:8088 or http://127.0.0.1:8088**. Note that the port number and IP address or hostname may be different depending on your installation method.

# Starting the Pentaho Enterprise Console in OS X

To start the console on OS X...

1. Open a command window.
2. Navigate to the `.../pentaho/server/enterprise-console` directory.
3. Type **./start-pec.sh** and press **RETURN**.
   Once the console is running, the following line appears in the command window: **Console is started. Console can be accessed using http://example.com:8088 or http://127.0.0.1:8088**.

# Stopping the Pentaho Enterprise Console

To stop the server...

1. Open a command window.
2. Go to the install directory that contains the file, **stop-pec.bat** (or **.sh**).
3. Type **stop-pec.bat** (or .**sh**)and press **ENTER** (or **RETURN**) to stop the console.

# Identifying Port Conflicts

The message, *Address already in use: JVM_Bind*, as the console starts, indicates that another program is using the port required by the Pentaho Enterprise Console (8088).

## Changing the Port Number

The Pentaho Enterprise Console starts up on port 8088. In most cases, you should not have to adjust your firewall configuration to accommodate this service. However, if you need to change the port number because of a conflict or port number restriction, edit the `/pentaho/server/enterprise-console/resource/config/console.properties` file to match your configuration.

# Accessing the Pentaho Enterprise Console

To access the console for the first time you must authenticate using default credentials:

1. Start the Pentaho Enterprise Console as described.
2. Go to **http://localhost:8088/** in your browser.
3. Enter the default user name, **admin**.
4. Enter the default password, **password**.
5. Click **OK**
   The console home page appears.

# Initial Pentaho Enterprise Console Configuration

Before you use the Pentaho Enterprise Console to manage the Data Integration Server, BI Server, or both, you must install license keys (referred to in the Enterprise Console interface as **subscription files**) and configure the console to locate certain files and directories. The first time you access the Pentaho Enterprise Console, the setup page will appear. If no license keys have been installed, all you will see is a license key dialog box.

👉 **Note:**

See *The Data Integration Server* for more information.

The Pentaho Enterprise Console displays this page any time the configuration is not correct, for example:

- The console configuration has been manually altered and is no longer valid
- The BI Server installation location has moved or been renamed
- The Enterprise Edition key has expired or is invalid
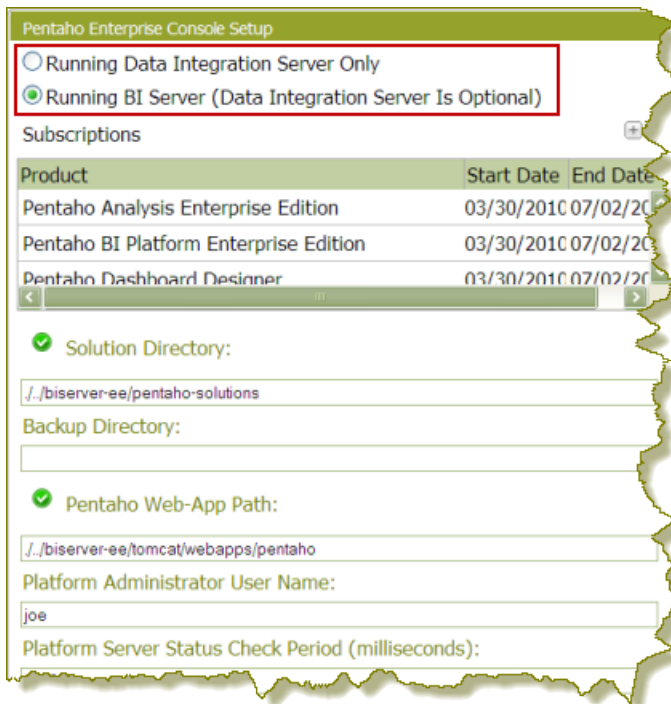


**Accessing the Configuration Page**

To access the configuration page at any time click 🛠 (setup) on the toolbar in the Pentaho Enterprise Console. The console setup page contains a list of Enterprise Edition license keys, plus the following settings:

| Setting | Description |
|---|---|
| **Solution Directory** | Paste the path to the solutions directory of the BI Server you want to administer into the **Solution Directory** text box; for example, `/pentaho/server/biserver-ee/pentaho-solutions/`. |
| **Backup Directory** | Paste the path to your backup directory in the Backup Directory text box. |
| **Pentaho Web-App Path** | Paste the path to the Web application directory of the BI Server you want to administer into the **Pentaho Web-App Path** text box, for example: `/pentaho/server/biserver-ee/tomcat/webapps/pentaho/` |
| **Platform Administrator User Name** | Enter the name of the administrator of the BI platform in the **Platform Administrator User Name** text box. |
| **Platform Server Status Check Period (milliseconds)** | Enter the status check period value in the **Platform Server Status Check Period** text box. This is the time period that |

| Setting | Description |
|---------|-------------|
|  | the console waits before pinging the BI Platform server for server status. The default value is provided but you can change it. |

👉 **Note:** For the Pentaho Enterprise Console to function correctly on a minimal basis, the solution directory and a valid Enterprise Edition key must be found.

# Installing or Updating an Enterprise Edition Key

You must install Pentaho Enterprise Edition keys associated with products for which you have purchased support entitlements. The keys you install determine the layout and capabilities of the Pentaho Enterprise Console, and the functionality of the BI Server and DI Server. Follow the instructions below to install an Enterprise Edition key through the Pentaho Enterprise Console for the first time, or to update an expired or expiring key. If you would prefer to use a command line tool instead, see *Appendix: Working From the Command Line Interface* on page 47.

👉 **Note:** If your Pentaho Enterprise Console server is running on a different machine than your BI or DI Server, you must use the command line tool to install and update license files; you will not be able to use the Pentaho Enterprise Console for this task.

👉 **Note:  License installation is a user-specific operation.** You must install licenses from the user accounts that will start all affected Pentaho software. If your BI or DI Server starts automatically at boot time, you must install licenses under the user account that is responsible for system services. If you have a Pentaho For Hadoop license, it must be installed under the user account that starts the Hadoop service as well as user accounts that run Pentaho client tools that have Hadoop functionality, and the account that starts the DI Server. There is no harm in installing the licenses under multiple local user accounts, if necessary.

1. If you have not done so already, log into the Pentaho Enterprise Console by opening a Web browser and navigating to `http://server-hostname:8088`, changing **server-hostname** to the hostname or IP address of your BI or DI server.
2. Click the **+** (plus) button in the upper right corner of the Subscriptions section.
   An **Install License** dialog box will appear.
3. Click **Browse**, then navigate to the location you saved your LIC files to, then click **Open**.
   LIC files for each of your supported Pentaho products were emailed to you along with your Pentaho Welcome Kit. If you did not receive this email, or if you have lost these files, contact your Pentaho support representative. If you do not yet have a support representative, contact the Pentaho salesperson you were working with.

   👉 **Note:** Do not open your LIC files with a text editor; they are binary files, and will become corrupt if they are saved as ASCII.

4. Click **OK**.
   The Setup page changes according to the LIC file you installed.

You can now configure your licensed products through the Pentaho Enterprise Console.

# Configuring the Proxy Trusting Filter

If you have set the BI Server to run on a specific IP address or hostname other than the default 127.0.0.1, you must modify the trusted proxy between the BI Server and Pentaho Enterprise Console to match that address or hostname.

1. Stop the Pentaho Enterprise Console.
2. Stop the BI Server.
3. Open `biserver-ee/tomcat/webapps/pentaho/WEB-INF/web.xml` and search for **TrustedIpAddrs**.

   👉 **Note:** The `param-value` immediately below **TrustedIpAddrs** is a comma-separated list of IP addresses that should be trusted.

```
<filter>
  <filter-name>Proxy Trusting Filter</filter-name>
```

```
  <filter-class>org.pentaho.platform.web.http.filters.ProxyTrustingFilter</filter-
class>
  <init-param>
    <param-name>TrustedIpAddrs</param-name>
    <param-value>127.0.0.1</param-value>
    <description>Comma separated list of IP addresses of a trusted hosts.</description>
  </init-param>
</filter>
```

4. Add the IP address of the host running the Pentaho Enterprise Console.

5. Open `enterprise-console/resource/config/console.xml` and search for **platform-username**.

6. Change the default **admin** user name, (default is joe) to your desired admin user name.

7. Start the BI Server.

8. Start the Pentaho Enterprise Console.

# The Data Integration Server

The Pentaho BI Suite installation provides you with a complete set of Business Intelligence tools, including the Data Integration server, a dedicated ETL server.

Among other things, the Data Integration server...

- Executes ETL jobs and transformations using the Pentaho Data Integration engine
- Allows you to manage users and roles (default security) or integrate security to your existing security provider such as LDAP or Active Directory
- Includes content management providing the ability to centrally store and manage your ETL jobs and transformations which encompasses full revision history on content and features such as sharing and locking for collaborative development environments
- Allows you to schedule activities and monitor scheduled activities on the Data Integration server from within designer environment

## Configuring the Enterprise Console

The Data Integration Server is an independent, separate server from the Pentaho BI Server. The Pentaho Enterprise Console can be used to manage the BI Server, the Data Integration server, or both. If you are setting up a dedicated server for ETL, follow the instructions below to configure the Pentaho Enterprise Console to display administrative functions associated with Pentaho Data Integration exclusively.

1. Click ![icon] (Setup) in the upper-right corner of the Pentaho Enterprise Console home page.
   The Pentaho Enterprise Setup page appears.
2. Enable **Running Data Integration Server Only**.
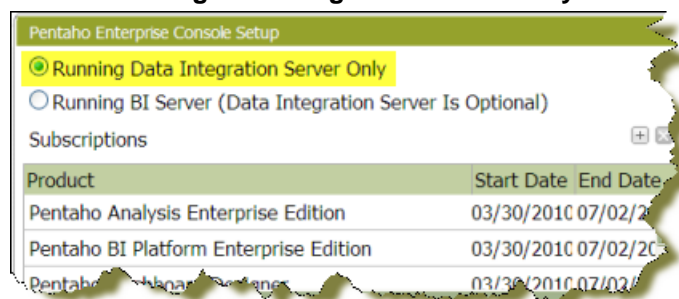


3. Click **OK** to exit the setup page.

The Pentaho Enterprise Console displays a page that contains information associated with Pentaho Data Integration. For more information, see *Getting Started with Pentaho Data Integration* available in the Pentaho Knowledge Base.

## Connecting to the Data Integration Server

You must register your Data Integration server to link it to the Pentaho Enterprise Console. After registration you can monitor activity associated with jobs and transformations being processed by the Data Integration server from the PDI dashboard in the Pentaho Enterprise Console.

To register your Data Integration server:

1. Make sure your Data Integration server is up and running.
2. In the Pentaho Enterprise Console home page, click **Pentaho Data Integration**.
3. Click to open the **Carte Configuration** dialog box.
4. In the **Carte Configuration** dialog box, enter the Data Integration server URL, (for example, http://localhost:9080/pentaho-di/), the server user name, and password.

   > **Note:** You can still configure the Pentaho Enterprise Console to connect to a basic Carte server using the appropriate URL; (for example, http://localhost:80/kettle/).

5. Click **OK** to complete the registration.
   A message appears if the connection to the Data Integration server is successful. If you see an error message, check to make sure your Data Integration server is running and that your access credentials are correct.

## Changing the Admin Credentials for the Pentaho Enterprise Console

The default user name and password for the Pentaho Enterprise Console are **admin** and **password**, respectively. You must change these credentials if you are deploying the BI Server to a production environment. Follow the instructions below to change the credentials:

1. Stop the Pentaho Enterprise Console.

   ```
   /pentaho/server/enterprise-console/stop-pec.sh
   ```
2. Open a terminal or command prompt window and navigate to the `/pentaho/server/enterprise-console/` directory.
3. Execute the **pec-passwd** script with two parameters: the Enterprise Console username and the new password you want to set.

   This script will make some configuration changes for you, and output an obfuscated password hash to the terminal.

   ```
   ./pec-passwd.sh admin newpass
   ```
4. Copy the hash output to the clipboard, text buffer, or to a temporary text file.
5. Edit the `/pentaho/server/enterprise-console/resource/config/login.properties` file with a text editor.
6. Replace the existing hash string with the new one, leaving all other information in the file intact.

   ```
   admin: OBF:1uo91vn61ymf1yt41v1p1ym71v2p1yti1ylz1vnw1unp,server-administrator,content-
   administrator,admin
   ```
7. Save and close the file, and restart Enterprise Console.

The Pentaho Enterprise Console password is now changed.

# Configuring Data Sources and Database Connections

Defining a database connection requires a JDBC class name for the database driver, a URL (server name, port number, database name) and the user ID and password needed to connect to your database. Out of the box, the Pentaho BI Suite installation comes with JDBC drivers listed below. The driver files are located in `.../pentaho/server/biserver-ee/tomcat/lib/`.

| MySQL | The location where solution definitions and the metadata they rely on is stored and maintained. Requests made to the platform to have actions executed rely on the action being defined in the Solution Repository. A *solution* consists of a collection of documents that define the processes and activities that are the system's part in implementing a solution to a business problem. These documents include action sequence definitions, workflow process definitions, report definitions, images, rules, queries, and more. |
|---|---|
| HSQLDB | The location where the sample data for the *Steel Wheels* solution is stored. HSQLDB is started when you execute the **start-pentaho.bat** script. |

JDBC connections, often referred to as **Pentaho Database Connections** or **named connections** are used by client tools, such as Pentaho Report Designer and Pentaho Design Studio. For example, when you create a report using Report Designer, you must reference the "connection name" to publish the report to the BI Server.

The **named connections** mentioned previously are *not* JNDI connections; however, you can create JNDI connections, if necessary. See *Adding a JNDI Data Connection to Tomcat* or *Adding a JNDI Data Connection to JBoss* for more information.

## Adding a Pentaho Database Connection

You must have the appropriate JDBC driver installed in Enterprise Console and the BI Server in order to establish a data source in Enterprise Console.

To define a Pentaho database connection:

1. In the Pentaho Enterprise Console go to **Administration** > **Database Connections**.
2. Click the **General** icon to display basic configuration options.
3. Click the plus sign (**+**) (add) if you cannot find your database in the default list.
   The **Add Database Connection** dialog box appears.
4. Type a connection name that concisely describes the data source.
5. Type or select the **Driver Class** from the list. The database driver name you select depends on the type of database you are accessing. For example, *org.hsqldb.jdbcDriver* is a sample driver name for a HSQLDB database.

   > **Note:** If your JDBC driver is not available, see *Adding a JDBC Driver*.

6. Type the **User Name** and **Password** required to access your database.
7. Type or select the **URL** from the list. This is the URL of your database; for example, *jdbc:hsqldb:hsql://localhost/sampledata*. JDBC establishes a connection to a SQL-based database and sends and processes SQL statements.
8. Click Test.
   A success message appears if the connection is established.
9. Click **OK** to save your entries.

## Adding a JDBC Driver

Before you can connect to a data source in any Pentaho server or client tool, you must first install the appropriate database driver. Your database administrator, CIO, or IT manager should be able to provide you with the proper driver JAR. If not, you can download a JDBC driver JAR file from your database vendor or driver developer's Web site. Once

you have the JAR, follow the instructions below to copy it to the driver directories for all of the BI Suite components that need to connect to this data source.

> **Note:** Microsoft SQL Server users frequently use an alternative, non-vendor-supported driver called JTDS. If you are adding an MSSQL data source, ensure that you are installing the correct driver.

**Backing up old drivers**

You must also ensure that there are no other versions of the same vendor's JDBC driver installed in these directories. If there are, you may have to back them up and remove them to avoid confusion and potential class loading problems. This is of particular concern when you are installing a driver JAR for a data source that is the same database type as your Pentaho solution repository. If you have any doubts as to how to proceed, contact your Pentaho support representative for guidance.

**Installing JDBC drivers**

Copy the driver JAR file to the following directories, depending on which servers and client tools you are using (Dashboard Designer, ad hoc reporting, and Analyzer are all part of the BI Server):

> **Note:  For the DI Server:** before copying a new JDBC driver, ensure that there is not a different version of the same JAR in the destination directory. If there is, you must remove the old JAR to avoid version conflicts.

* **BI Server:** `/pentaho/server/biserver-ee/tomcat/lib/`
* **Enterprise Console:** `/pentaho/server/enterprise-console/jdbc/`
* **Data Integration Server:** `/pentaho/server/data-integration-server/tomcat/webapps/pentaho-di/WEB-INF/lib/`
* **Data Integration client:** `/pentaho/design-tools/data-integration/libext/JDBC/`
* **Report Designer:** `/pentaho/design-tools/report-designer/lib/jdbc/`
* **Schema Workbench:** `/pentaho/design-tools/schema-workbench/drivers/`
* **Aggregation Designer:** `/pentaho/design-tools/agg-designer/drivers/`
* **Metadata Editor:** `/pentaho/design-tools/metadata-editor/libext/JDBC/`

> **Note:** To establish a data source in the Pentaho Enterprise Console, you must install the driver in both the Enterprise Console and the BI Server or Data Integration Server. If you are just adding a data source through the Pentaho User Console, you do not need to install the driver to Enterprise Console.

**Restarting**

Once the driver JAR is in place, you must restart the server or client tool that you added it to.

**Connecting to a Microsoft SQL Server using Integrated or Windows Authentication**

The JDBC driver supports Type 2 integrated authentication on Windows operating systems through the **integratedSecurity** connection string property. To use integrated authentication, copy the **sqljdbc_auth.dll** file to all the directories to which you copied the JDBC files.

The **sqljdbc_auth.dll** files are installed in the following location:

```
<installation directory>\sqljdbc_<version>\<language>\auth\
```

> **Note:** Use the **sqljdbc_auth.dll** file, in the x86 folder, if you are running a 32-bit Java Virtual Machine (JVM) even if the operating system is version x64. Use the **sqljdbc_auth.dll** file in the x64 folder, if you are running a 64-bit JVM on a x64 processor. Use the **sqljdbc_auth.dll** file in the IA64 folder, you are running a 64-bit JVM on an Itanium processor.

## Adding a Simple JNDI Data Source

Rather than add individual JDBC data source connections to each Pentaho client tool and server, sometimes it's easier to establish a single global JNDI connection that can be used across all Pentaho BI Suite products. JNDI connections can be configured at the application server level if you want to share them among several deployed applications; to learn how to do this you should consult your application server documentation. Pentaho provides an alternative method for defining a JNDI connection that exists only for its locally installed client tools and servers. Follow the directions below to establish a simple JNDI connection.

1.  Navigate to the **.pentaho** directory in your home or user directory.

    For a user name of **fbeuller**, typically in Linux and Solaris this would be `/home/fbeuller/.pentaho/`, and in Windows it would be `C:\Users\fbeuller\.pentaho\`

2.  Switch to the `~/.pentaho/simple-jndi/` subdirectory. If it does not exist, create it.

3.  Edit the **default.properties** file found there. If it does not exist, create it now.

4.  Add a data source by declaring a JNDI name followed by a forward slash, then a JNDI parameter and its proper value.

    Refer to *Simple JNDI Options* on page 14 for more information on parameter options.

    ```
    SampleData/type=javax.sql.DataSource
    SampleData/driver=org.hsqldb.jdbcDriver
    SampleData/user=pentaho_user
    SampleData/password=password
    SampleData/url=jdbc:hsqldb:mem:SampleData
    ```

5.  Save and close the file.

You now have a global Pentaho data source that can be used across all of the client tools and servers installed on this machine. You must restart any running Pentaho program in order for this change to take effect.

## Simple JNDI Options

Each line in the data source definition must begin with the JNDI name and a forward slash (/), followed by the required parameters listed below.

| Parameter | Values |
|---|---|
| type | **javax.sql.DataSource** defines a JNDI data source type. |
| driver | This is the driver class name provided by your database vendor. |
| user | A user account that can connect to this database. |
| password | The password for the previously declared user. |
| url | The database connection string provided by your database vendor. |

```
SampleData/type=javax.sql.DataSource
SampleData/driver=org.hsqldb.jdbcDriver
SampleData/user=pentaho_user
SampleData/password=password
SampleData/url=jdbc:hsqldb:mem:SampleData
```

# Adding an Advanced Pentaho Database Connection

To add an advanced Pentaho Database connection:

1.  In the Pentaho Enterprise Console go to **Administration** > **Database Connections**.

2.  Click the **Advanced** icon to display basic advanced options.

3.  Enter the maximum number of active instances, (**Max Active Conn**), that can be allocated from this pool at the same time.

4.  Enter the maximum number of connections that can sit idle (**# Idle Conn**) in this pool at the same time.

5.  Enter a **Validation Query**. This SQL query that can be used by the pool to validate connections before they are returned to the application. If specified, this query must be an SQL SELECT statement that returns at least one row.

6.  Enter the maximum number of milliseconds that the pool will "wait" (when there are no available connections) for a connection to be returned before throwing an exception.

7.  Click **Test**.
    A success message appears if the connection is established.

8.  Click **OK** to save your entries.

# Adding a JNDI Data Connection to Tomcat

If you would prefer to use a JNDI connection to your database instead of the standard Pentaho method, follow the instructions below:

1. Consult your database documentation to determine the class name and connection string for your database.
2. Stop the Tomcat server by running the **stop-pentaho** script.
3. Edit the `/pentaho/server/biserver-ee/tomcat/conf/server.xml` with a text editor.
4. Anywhere inside of the <host> element, add the XML snippet shown below:

```
<Context path="/pentaho" docbase="webapps/pentaho/">
    <Resource name="jdbc/myDataSource" auth="Container" type="javax.sql.DataSource"
    factory="org.apache.commons.dbcp.BasicDataSourceFactory" maxActive="20"
 maxIdle="5"
    maxWait="10000" username="dbuser" password="password"
 driverClassName="org.postgresql.Driver"
    url="jdbc:postgresql://127.0.0.1:5432/myDataSource" />
</Context>
```

The above example shows a simple PostgreSQL configuration. Replace the **Resource name**, **username**, **password**, **driverClassName**, and **url** parameters (or any relevant connection settings) to match your connection information.

5. Save and close the `server.xml` file.
6. Start the Tomcat server by running the **start-pentaho** script.

Tomcat can now properly connect to your database.

# Adding a JNDI Data Connection to JBoss

If you must use JNDI connections instead of Pentaho-managed connections for your databases, follow the configuration instructions below. The preferred and much simpler method of database connection management is through the Pentaho Enterprise Console, but there may be instances where you want to share data connection settings with other Web applications.

1. Consult your database documentation to determine the class name and connection string for your database.
2. Stop the JBoss server.
3. Now you need to specify a database and set a JNDI name to reference it by. These properties are specified in a data source XML file that you will create specifically for your database. Navigate to the `/jboss/server/default/deploy/` directory.
4. Open a text editor and create a file named **my_datasource-ds.xml**, and use this example code as a basis for its content, changing the database type, address, port, and database name, the driver class for your database (if it isn't MySQL, as in the example), and the access credentials:

```
<?xml version="1.0" encoding="UTF-8"?>
<datasources>
    <local-tx-datasource>
        <!-- JNDI name of the datasource, it is prefixed with java:/ -->
        <jndi-name>my_jndi_datasource</jndi-name>
        <connection-url>jdbc:mysql://localhost:3306/myDatabaseName</connection-url>
        <driver-class>org.gjt.mm.mysql.Driver</driver-class>
        <user-name>username</user-name>
        <password>password</password>
    </local-tx-datasource>
</datasources>
```

5. Next, edit the `/jboss/server/default/deploy/pentaho.war/WEB-INF/web.xml` file and search for this comment:

```
<!-- insert additional resource-refs -->
```

6. Below the comment, create a new resource-ref that follows this example, changing the necessary details as you did previously (the res-type remains the same for all databases, so you do not need to modify that property):

```
<resource-ref>
```

```
    <description>my_jndi_datasource</description>
    <res-ref-name>jdbc/my_jndi_datasource</res-ref-name>
    <res-type>javax.sql.DataSource</res-type>
    <res-auth>Container</res-auth>
</resource-ref>
```

**7.** Now edit the `jboss-web.xml` file in the same directory, and search for this comment:

```
<!-- insert additional jboss resource-refs -->
```

**8.** Below the comment, create a new resource-ref that follows this example, again changing the necessary details to match your database:

```
<resource-ref>
    <res-ref-name>jdbc/my_jndi_datasource</res-ref-name>
    <res-type>javax.sql.DataSource</res-type>
    <jndi-name>java:/my_jndi_datasource</jndi-name>
</resource-ref>
```

**9.** Start the JBoss server.

You should now have access to your database through the BI Server. If you have not already prepared your data for reporting and analysis through Metadata Editor and Schema Workbench, you should do so now if you intend to use any elements of Pentaho Reporting and Analysis.

# Adding a Data Source in the Pentaho User Console

The Data Sources feature allows you to connect to your data quickly so that the data can be used to create reports, (such as ad hoc reports, dashboards, and analyzer reports), in the Pentaho User Console. For example, when CSV data is uploaded to the Pentaho BI Server metadata models are automatically generated for each data source. The models can be further customized to adjust field names, formatting, aggregation types, and dimensional hierarchies (for Pentaho Analysis).

If you are running a test environment, you can quickly add, edit, and delete a relational or CSV data source in the Pentaho User Console using the Data Source wizard. You can restrict access to a data source by individual user or user role; by default, authenticated users are limited to view-only permissions. (See "Assigning Data Source Permissions for the Pentaho User Console" in the **Pentaho Security Guide** for more information.) This means that they are limited to selecting a data source from a list of available options.

> **Note:** Non-administrative users of the Pentaho User Console do not see icons or buttons associated with adding, editing, or deleting a data source. They are also not allowed to create a data source.

When selecting a relational data source, users can query against the data source within the scope of data defined in query you set up. If users choose a CSV-based data source, data is loaded from the CSV file and staged on the Pentaho BI Server.

The **Data Source Wizard**, which allows you to set up access to a relational data source or CSV data source in the Pentaho User Console, is available in the following instances:

- When you click the **New Data Source** button on the Pentaho User Console Quick Launch page
- When you select **File** -> **New** -> **Data Sources** in the Pentaho User Console menu bar
- When you select **File** -> **Manage** -> **Data Sources** from the Pentaho User Console menu bar and click  (Add)
- When you click **Add** in the **Select a Data Source** step in Ad Hoc Reports
- When you click **Create Data** in charts in the Dashboard
- When you click  (Add) as a first step of the create **Data Table** process
- When you click **Create New Data Source** when creating a new Analyzer report

### Using Pentaho Metadata Editor

In a production environment, users of the Pentaho User Console get access to data for creating ad hoc reports and building dashboards from metadata models provided by an administrator. Using Pentaho Metadata Editor (PME), you can restrict what data a user can access down to a table, column, or row. This type of security fine-tuning, and many other capabilities, are not provided by the Data Sources feature. The default metadata models created when adding new data sources in the Pentaho must be opened in Pentaho Metadata Editor for further customization.

For more information, refer to the *Pentaho Metadata Editor Guide* available in the Pentaho Knowledge Base.

## Creating a Relational Data Source in the Pentaho User Console

Follow the instructions below to create a relational data source. In addition to allowing you to connect to your database (using JDBC), you must also enter a SQL statement that defines the "scope, "(similar to a database view), of your data source.

> **Note:** In the context of the Pentaho User console, a relational "data source" is a combination of a connection plus a query. Pentaho recommends that you use the Pentaho Metadata Editor to create relational data sources for a production environment.

You can customize how columns are presented to users who are building queries against the new data source; for example, you can define "friendly" column names and select options that define how data is aggregated (sum, min., max., etc.), and more.

Once you create the data source, it is available to users of the Dashboard Designer, Chart Designer, and Ad Hoc Report.

You must be logged on to the Pentaho User Console as an administrator to follow the instructions below. Additionally, your database must be up and running in order for you to complete the steps required in the Data Source Wizard.

1. In the **Data Source Wizard**, type a name for the new data source in the **Data Source Name** field.

   👉 **Note:** The name of the data source is important because it becomes the name of the data source displayed in Analyzer, Ad Hoc Report, and Chart Designer.

2. Under **Source Type**, select **Database** as your data source type.

3. If not already configured, click ⊕ (Add) to add a connection to the database that contains the content you want to access. The **Database Connection** dialog box appears.
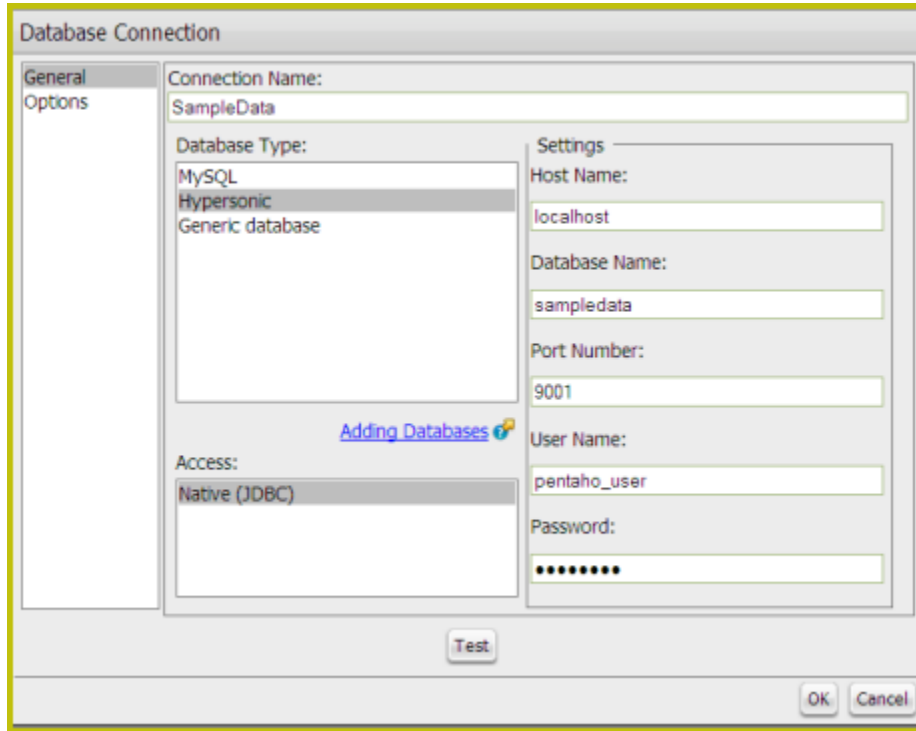


   👉 **Note:** If this is the first time you are using the wizard, you will not see a list of connections under Available Connections. When you have added your connection, the connection name will appear in the list. Click the **Adding Databases** hyperlink if you need help connecting to your database.

4. In the **Connection Name** text box, enter an easy-to-remember name that identifies the data you are accessing.

5. Select a **Database Type** from the list.

   Text fields for required settings associated with your connection type appear under **Settings** on the right. See *Adding a JDBC Driver* if the driver you need is not listed.

6. Enter the appropriate connection information for your database type.

   Consult the *list of JDBC class names* for more information.

7. In the Connection dialog box, click **Test** to test your connection and click **OK**.
   A success message appears if your settings are correct. The connection name appears in the list.

8. Under **SQL Query**, enter your SQL Query. Click **Preview** to ensure that your query returns data.
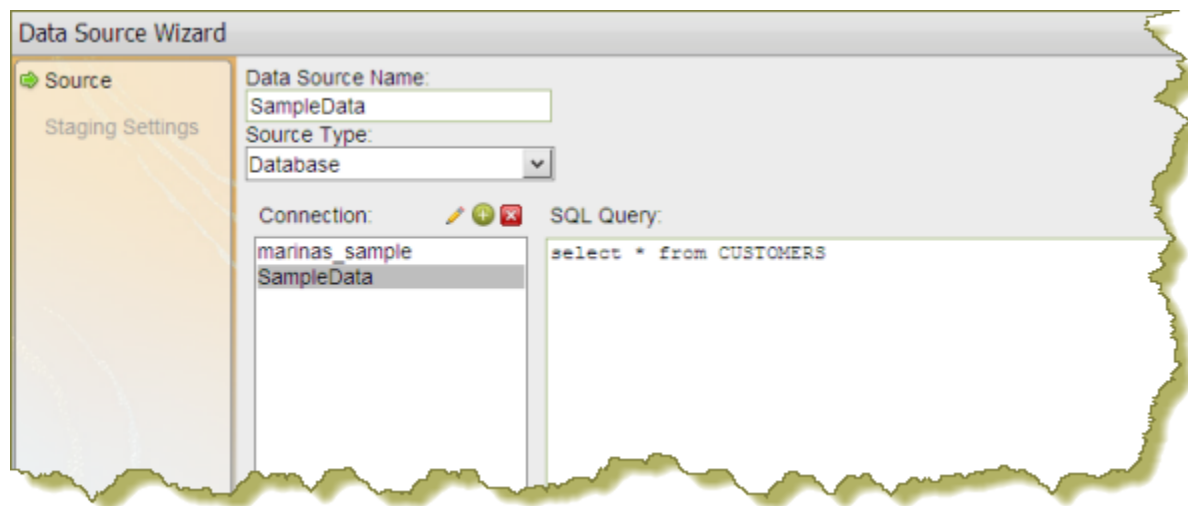
9. Click **Close** to exit the Preview window.

10. Click **Finish**.

    In the background, after you click **Finish**, a default metadata model is generated based on the query for use in Analyzer, Ad Hoc Reporting, Report Designer, and Chart Designer. See *Editing a Data Source Connection* for instructions about changing connection details.

# Creating a CSV Data Source in the Pentaho User Console

Follow the instructions below to create a CSV-based data source. For testing purposes, you can save an Excel spreadsheet as a CSV file.

1. In the **Data Source Wizard**, type a name that identifies your new data source in the **Data Source Name** field.

2. Under **Source Type** select **CSV** from the list to define a CSV file as your data source and click **Next**.

3. Click **Import** to open the **Import File** dialog box.

4. Click **Browse** to search for your CSV file; click **Import** when you locate the file.

    The Import File dialog box closes and the CSV file is uploaded to the BI Server. You can click **Cancel**, if necessary, and select different file.

    > **Note:** Notice that a preview of the contents of your file appears in the lower portion of the wizard page. If items are not aligning correctly, it is because you must specify your delimiter and enclosure types. Go on to the next step.

5. Specify the delimiter and enclosure types. Enable **First row is header** if the first row of your CSV file is a heading for columns in the file.
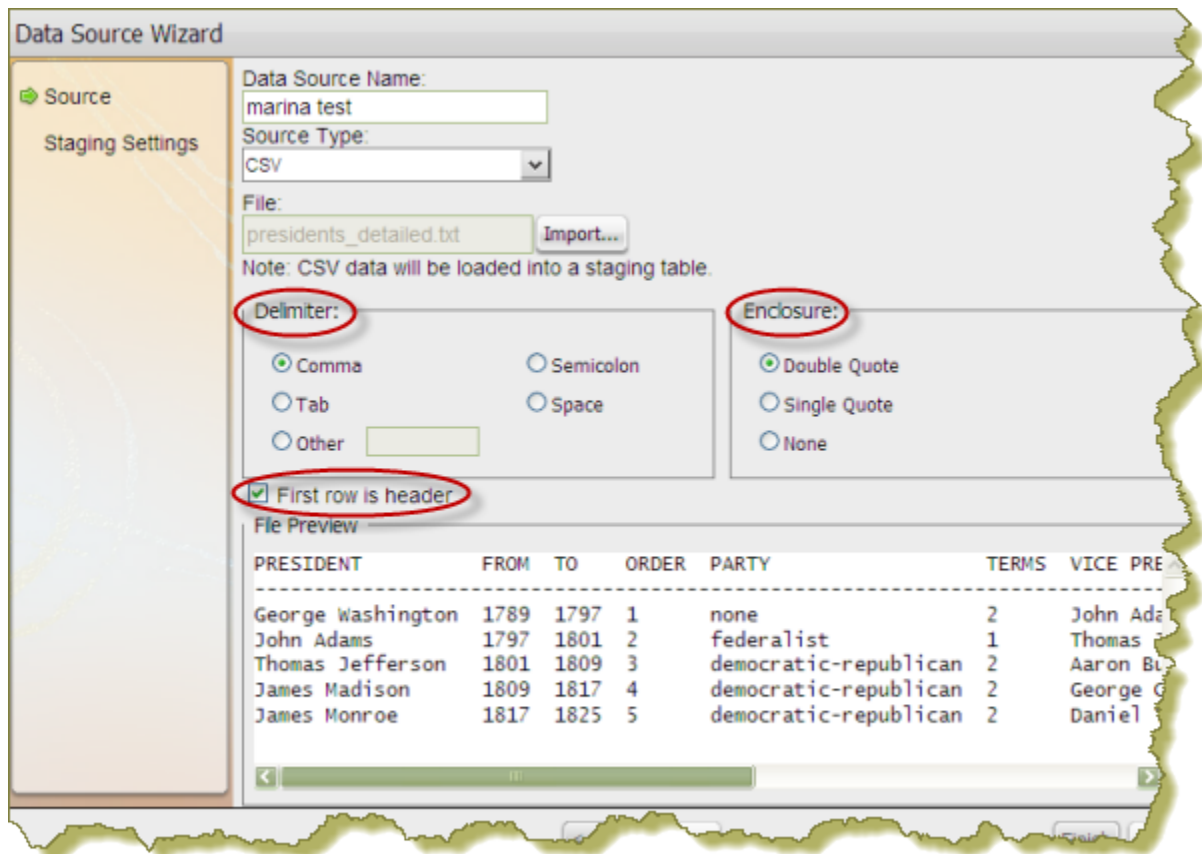
> **Important:** You must make selections under **Delimiter** and **Enclosure** for the Metadata table to build correctly. In the example line, `1,4,5,02/12/77,"String Value"` the commas are delimiters and the quotation marks around "String Value" is the enclosure.

The **File Preview** window displays the first few lines of your CSV file based on the selections you made for specifying the delimiter, enclosure, and header. Once the columns align correctly in the preview you will know that the delimiter and enclosure have been set correctly.

6. Click **Next** to display more details about the columns in your source file and to edit staging-related settings.

> **Note:** Notice that all columns are enabled in this step. Click **Deselect All** to disable all columns. You can then select the columns that require formatting individually. Click **Select All** to enable all columns again, if necessary.

7. If applicable, change the **Name** and **Type** values. Use the drop-down list to display format options for dates and numeric values. Formats are auto-detected based on the contents of your data source; however, if you see a format option that is not correct, enter it manually in the appropriate **Source Format** text box.

> **Note:** Refer to *Sample Date Formats* for additional help formatting dates. Refer to *Decimal Formats* for additional help formatting numbers.

**Length** indicates the maximum number of characters allowed in a field. **Precision** is the number of digits after a decimal point. Drop-down lists are not enabled for certain data types such as the String data type.
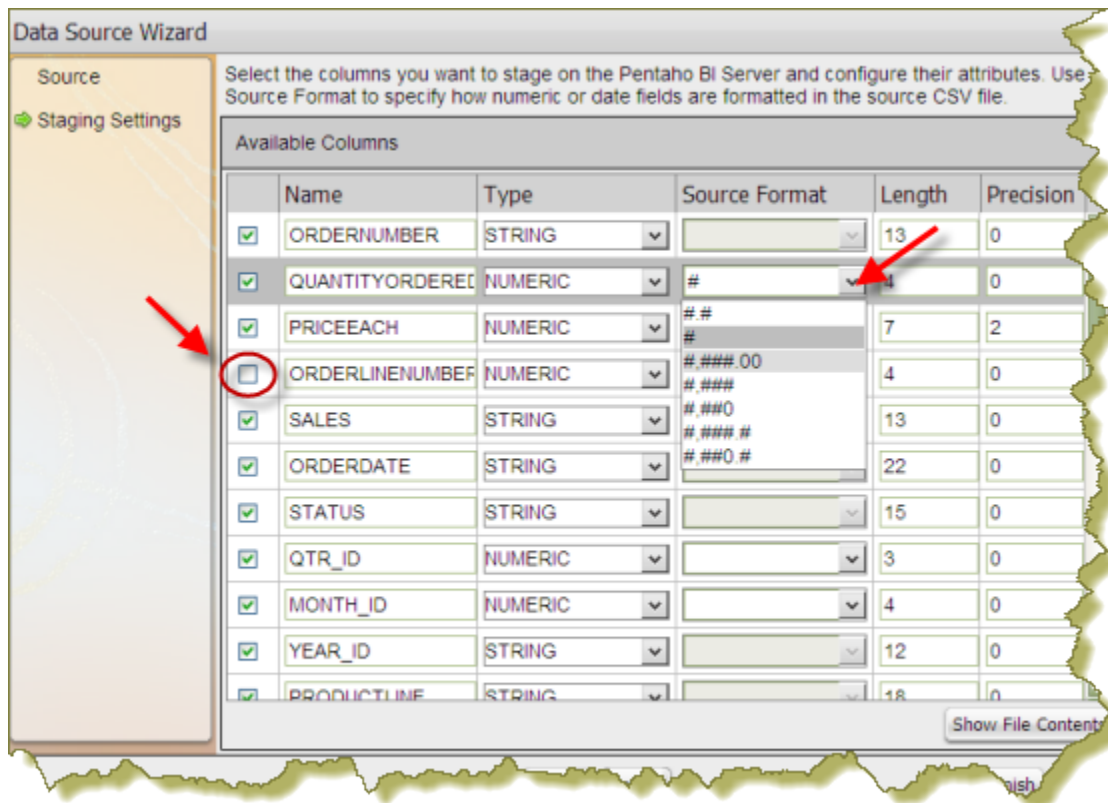
> **Important:** Boolean values are rendered as "true" or "false."

8. Click **Show File Contents** to see a sample of the data in your source file. Click **OK** to return to staging and make changes if necessary.

9. Click **Finish** when you have completed your configuration and formatting tasks.
A success message appears indicating the number of rows being loaded into the database and that a default metadata model was created. Enable **Keep default model** to use the default model or click **Customize model now** to customize the model.

The data in your CSV file is extracted and staged temporarily in a database table. Your new data source is now available to users. To refine the data and make it more accessible to users of reports, dashboards, and data tables, you must use the Data Source Modeler. See *Editing a Data Source in the Pentaho User Console*.

## Increasing the CSV File Upload Limit

Follow the instructions below if you need to increase the size of the upload limit associated with CSV files.

1. Go to `...\biserver-ee\pentaho-solutions\system` and open the **pentaho.xml** file

2. Edit the XML as needed; (sizes are measured in bytes):

```
<file-upload-defaults>
      <relative-path>/system/metadata/csvfiles/</relative-path>

      <!-- max-file-limit is the maximum file size, in bytes, to allow to be uploaded
  to the server -->
      <max-file-limit>10000000</max-file-limit>

      <!-- max-folder-limit is the maximum combined size of all files in the upload
  folder, in bytes. -->
      <max-folder-limit>500000000</max-folder-limit>

</file-upload-defaults>
```

3. Save your changes to the file.

4. In the **Pentaho User Console**, go to **Tools** -> **Refresh System Settings** to ensure that the change is implemented.

## Changing the Staging Database

Hibernate is the default staging database for CSV files. Follow the instructions below if you want to change the staging database.

1. Go to `...\biserver-ee\pentaho-solutions\system\data-access` and open **settings.xml**.
2. Edit the settings.xml file as needed. The default value is shown in the sample below.

```
<!-- settings for Agile Data Access -->
<data-access-staging-jndi>Hibernate</data-access-staging-jndi>
```

> 👉 **Note:** This value can be a JNDI name or the name of a *Pentaho Database Connection*.

3. Save your changes to the **settings.xml** file.
4. Restart the Pentaho User Console.

## Where Models are Stored

All generated models are stored in the following location, `\pentaho-solutions\admin\resources\metadata`.

> 👉 **Important:** Pentaho recommends that you *do not* change the location where models are stored.

# Using the Modeler to Edit a Data Source

You can edit a data source in Ad Hoc Reports, Pentaho Analyzer, and in the Dashboard. If you are editing a data source while using Ad Hoc Reports, select the data source you want to edit from the list of available data sources and click **Edit**. If you are creating a chart or data table, select the data source you want to edit from the list and click ✎ (Edit). If you have created an Analyzer report, click ✎ (Edit) in the upper right corner next to **Available Fields**. You can also edit a data source by navigating to **File** -> **Manage** -> **Data Sources** in the Pentaho User Console menu bar and opening the **Select Data Source** dialog box.
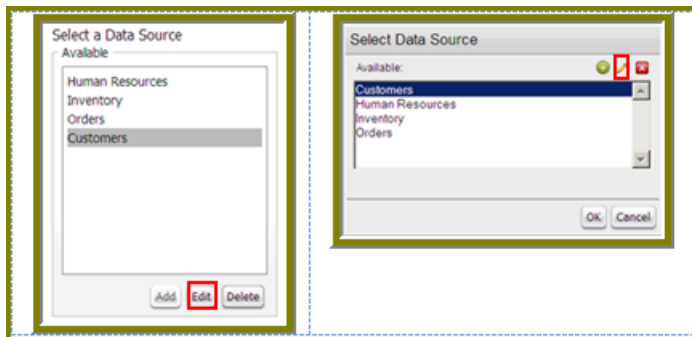


> 👉 **Note:** If you want to edit the connection information, see *Editing a Data Source Connection in the Pentaho User Console*.

When you edit a data source you are using the Data Source Modeler. The modeler allows you to design reporting and OLAP metadata models on the fly. The modeler also allows you to refine and organize your data so that it is more accessible to report users. For example, you can change the field display names, set an aggregation type, and change formatting,

Follow the instructions below to edit a data source:

1. In the **Data Source Modeler**, examine your data.
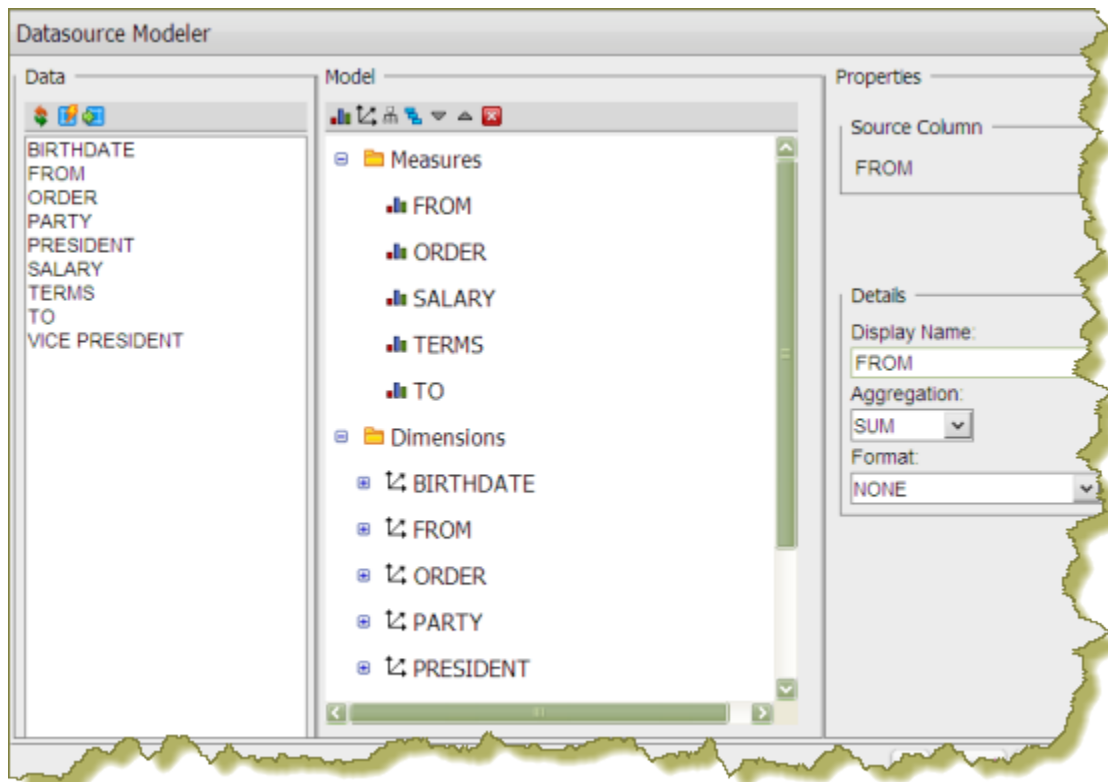
Measures are numeric values such as sales price, cost, and profit. **Dimensions** are the context that allow users to understand the significance of the measures. For a sales database, the dimensions could include Product, Store, Salesperson and so on. In other words, users can examine the sales figures by store, by product, and by sales.

Dimensions can have one or more hierarchies (⬡); (hierarchies are not shown in the example). Hierarchies are levels that organize data into a logical structure. Click **(+)** next to fields under Measures and Hierarchies to display their contents. Moving up and down between the levels of a hierarchy is called *drilling*.

You can also edit the source data to define the properties of your CSV file or re-enter a SQL query. In the example above, the FROM, TO, TERMS, and ORDER fields must be redefined as string types, not numeric. The only

applicable measure in this example, is SALARY. Unnecessary fields can be removed clicking ❌ (Delete) in the modeler toolbar.

You can also move a field under Measures or Dimensions by clicking and dragging the field into its appropriate location. A green check mark indicates that you can place your field without error.

👉 **Note:** Click 🔘 (Clear Model, in the upper right corner) to clear the existing model and start over.
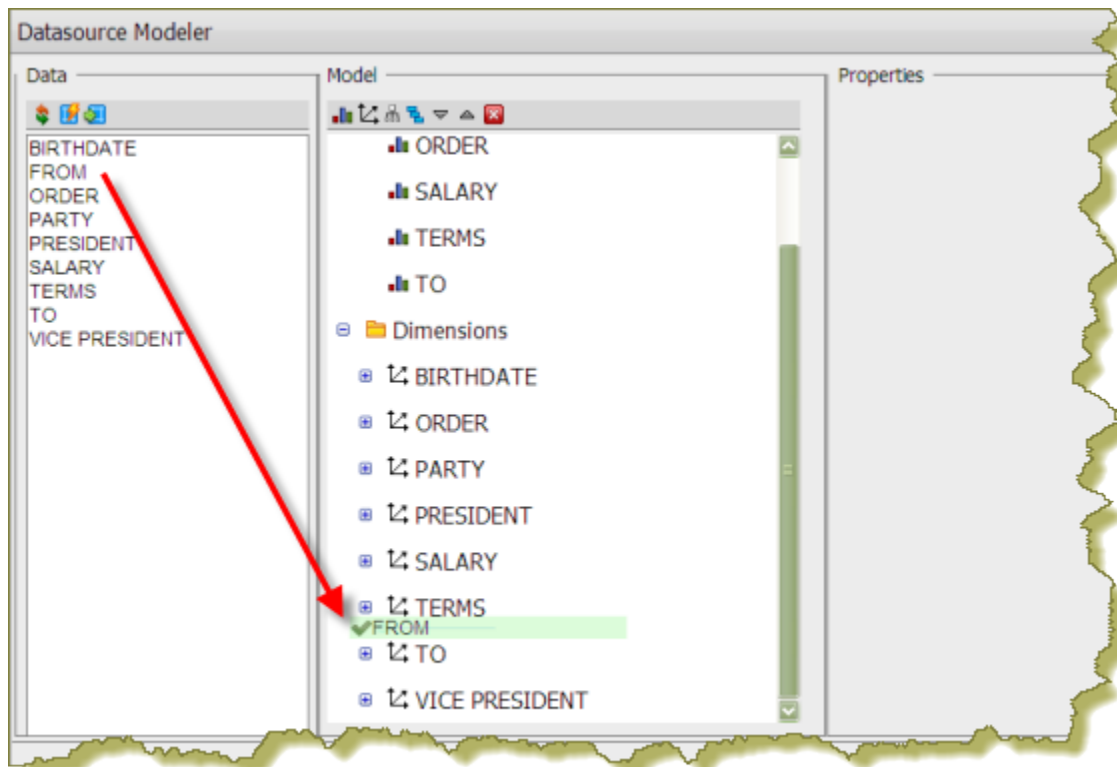
2. If you have clicked **Edit Source** in the modeler to make changes to a SQL query or CSV flat file (delimiter, enclosure, or header), click **Finish** to update the metadata table.

   If you disable columns that were previously enabled, a prompt appears in the modeler indicating that the data has been deleted. You must adjust the data in the modeler before proceeding.

3. Click **OK** to update the data source.

The updated data source is now available to users.

## Editing a Data Source Connection in the Pentaho User Console

Follow the instructions below to edit a data source connection in the Pentaho User Console:

1. In the **New Data Source** dialog box, under **Connection**, select the connection you want to edit.

2. Click 
   The Database Connection dialog box appears.

3. Update the connection information as needed. You can click **Test** to make sure your connection is working correctly.

4. Click **OK** to exit the Database Connection dialog box.

## Deleting a Data Source Connection in the Pentaho User Console

Follow the instructions below to delete a data source connection associated with the Pentaho User Console.

> **Important:** If you delete a data source connection, all reports, charts, and other content users have created using the connection will no longer render.

1. Under **Connection** in the New Data Source dialog box, select the connection you want to delete.

2. Click (Delete).
   A warning dialog box appears.

3. Click **OK** to delete the connection.

# Deleting a Data Source in the Pentaho User Console

If you have administrative permissions, you can delete data sources associated with ad hoc reports and dashboards in the Pentaho User Console.

**Important:** When you delete a data source all reports or charts that end-users have created based on that data source will no longer render. Deleting a data source also deletes the model associated with that data source.

The delete Data Source option is available in the following instances:

- When you select a data source and click ![Delete] (Delete) when in the **Select Data Source** dialog box available from **File** -> **Manage** -> **Data Sources**
- When you select a data source and click **Delete** in the **Select Data Source** step in Ad Hoc Reports
- When you select a data source and click ![Delete] (Delete) when creating a chart
- When you select a data source and click ![Delete] (Delete) when creating a data table

# Managing Schedules

The Scheduler allows you to create, update, delete, run, suspend, and resume one or more schedules, (Public and Private), in the BI Platform. In addition, you can suspend and resume the Scheduler itself. In the context of the BI platform, a schedule is a time (or group of times) associated with an action sequence (or group of action sequences). In many instances, the output of an action sequence associated with a Public schedule is a report; for example, a sales report to which a manager or salesperson can subscribe. As the administrator, the schedule (or schedules) you designate determines when the Scheduler allows the action sequence to run. Private schedules are ad hoc, non-subscription schedules, which are associated with one action sequence only.

> **Note:** Public Schedules were formerly called, "subscription schedules;" private schedules were formerly called, "regular schedules."

In addition to associating a time (or group of times in the case of a repeating schedule) with an action sequence (or group of action sequences), the public schedule is also associated with a user's My Workspace. When an action sequence runs on its defined schedule, the output of the action sequence (typically a report) is archived in the My Workspace of the user(s) who have subscribed to that action sequence. This allows the subscribers to view the output of the action sequence (the report) at any time following its execution.

Why not allow BI Platform users to create schedules whenever they want? Allowing that much flexibility may, among other things, overload servers. In most instances, you know when it is most sensible to schedule an action sequence to run; for example, after all stores upload their sales figures. In other instances, sales data may not change for a week or month, so reporting hourly would not make sense. You, or a solution developer, can define as many schedules as needed for a specific action sequence. Users are allowed to choose from schedules that make sense to them and you can schedule the run to occur at a time of minimal load.

## Entering Schedules in the Schedule Creator Dialog Box

Enter schedules associated with your action sequences in the Schedule Creator dialog box. The Schedule Creator makes it easy for you to enter schedules without having to learn the syntax of Cron expressions; however, it provides you with the option to enter Cron expressions if that is your preference.

The scheduling process is not complete until you associate the schedule with one or more action sequences. If you are creating a private schedule, you are limited to associating the schedule to a single action sequence.

To enter schedules...

1. In the main page of the Pentaho Enterprise Console, click **Administration**.
2. Click the **Scheduler** tab.
3. In the Scheduler, click ☐ (New) to open the **Scheduler Creator** dialog box.

   > **Note:** By default, the Scheduler creates a public schedule. To create a private schedule, disable the **Make Public** check box.

4. Under Schedule, enter a **Name** for the schedule, for example, Monthly Sales.
5. Enter a **Group** associated with the schedule, for example, Sales Schedules.
6. Enter a short **Description** of the schedule. for example, "Schedule runs on the first of each month, schedule runs on Monday of each week."
7. Select a **Recurrence Type**. You can schedule the action sequence to run once at a particular date and time only, or have it recur in seconds, minutes, hours, daily, weekly, monthly, yearly, or recur based on a Cron string. The options in the Recurrence Editor change depending on the type of recurrence you select.

   > **Note:** You can use the Schedule Creator to enter a Cron expression manually by selecting Cron from the Recurrence Type list. See *CRON Expressions in Detail* (link below) to learn more about Cron expressions.

   You must now associate the new schedule with one or more action sequences.

## Associating Schedules with Action Sequences

After you add your schedules, you must associate them with action sequences.

**Caution:** For private schedules, you must associate a schedule with a minimum of one action sequence or an error message appears.

To associate schedules with action sequences...

1. In the **Schedule Creator** dialog box, click the **Selected Files** tab.
2. Click the Add button next to **Associate this public schedule with the selected files** to open the **Select** (file chooser) dialog box.
3. In the **Select** dialog box double-click a folder to display contents or select the folder and click **Open**.
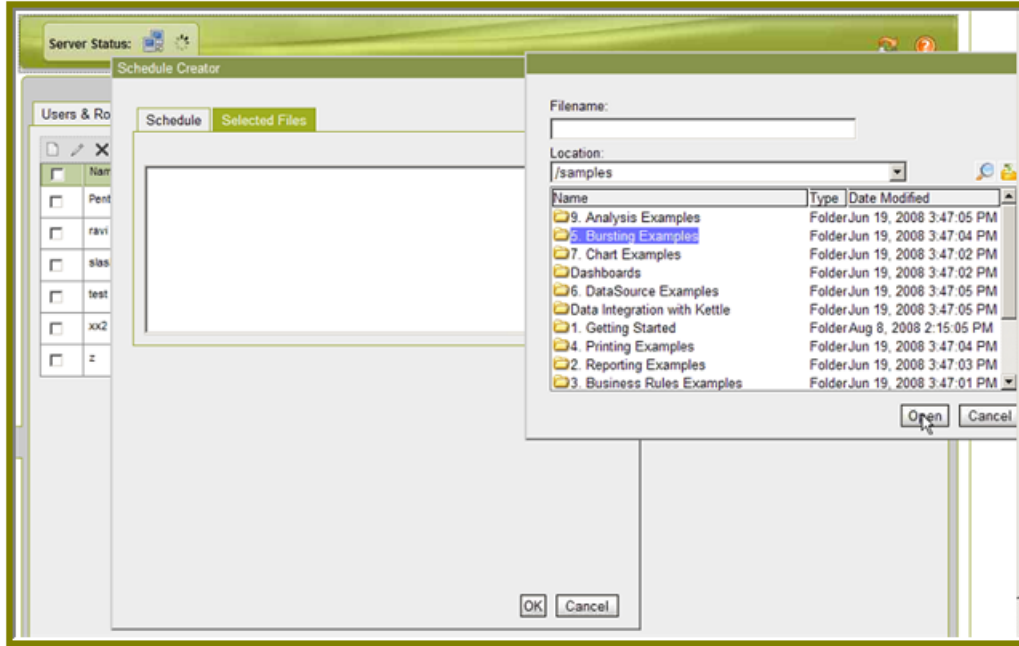


4. Choose the action sequence file you want and click **Select**.
   The action sequence file appears under **Selected Files**. You can add more action sequence files as needed by repeating the process above.
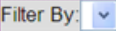
## Removing Action Sequence Files from a List

A schedule can be associated multiple action sequence files. To remove an action sequence file from a list, select the file and click ⬛ (minus). To remove multiple action sequence files, press **CTRL + CLICK** to select the files for deletion.

## Examining the List of Schedules

As you create new schedules, the schedules appear in a list box. By examining the list, you can identify the **Name** and **Group** associated with each schedule. You can also determine the status (**State**) of each schedule and read a brief description of the schedule. In addition, you can determine when the schedule was first run (**Fire Time - Last/Next**) and when it will run again. The controls on the top corners of the Scheduler page define the tasks you can perform using the Scheduler:

| Icon | Control Name | Function |
|------|--------------|----------|
| | **Create Schedule** | Allows you to create a new schedule |
| | **Edit Schedule** | Allows you to edit the details of a schedule |
| | **Delete Schedule** | Allows you to delete a specified schedule; however, if the schedule is currently executing in a scheduler thread it continues to execute but no new instances are run<br><br>**Note:** To reliably delete a public schedule, (in the |

| Icon | Control Name | Function |
|------|-------------|----------|
| | | Pentaho User Console), users must suspend the schedule first then delete it from their workspace. (Users must have administrative privileges to delete a schedule.) After the delete process completes, they may restart the schedule. |
| | **Suspend Schedule** | Allows you to pause a specified schedule; once the job is paused the only way to start it again is with a Resume Schedule |
| | **Resume selected Schedule(s)** | Allows you to resume a previously suspended schedule; once the schedule is resumed the Scheduler applies misfire rules if needed |
| | **Run selected Schedule(s) Now** | Allows you to run a selected schedule(s) at will |
| | **Refresh** | Allows you to refresh the list of schedules |
| Filter By: | **Filter by** | Allows you to search for a specific schedule by group name |

# Configuring the BI Server

This section contains instructions for configuring and testing your BI server settings.

## Software Updates

To configure your software updates...

1.  In the Pentaho Enterprise Console home page, click **Configuration**.
2.  Click the **General** tab.
3.  In the **Global Settings** group box, update the version-related settings as shown below:

| Item | Description |
|---|---|
| Version Check Interval | Interval, in seconds, to check for software updates |
| Version Check Flags | Flags representing the release types in which you are interested such as major or minor version updates |
| Disable Version Check | Disables the version check |

4.  Click **Submit** when you have completed your changes to these settings.
    A success message appears.

## Email Settings

To configure email settings:

1.  In the Pentaho Enterprise Console home page, click **Configuration**.
2.  Click the **BI Components** tab
3.  Enter in your settings as described below:

| Setting | Description |
|---|---|
| Use Authentication | Enable if the email server requires the sender to authenticate. |
| Use Start TLS | Disabled in most instances except when using Gmail. Enables the use of the STARTTLS command (if supported by the server) to switch the connection to a TLS-protected connection before issuing any logon commands. Note that an appropriate trust store must configured so that the client will trust the server's certificate. Defaults to disabled. |
| Default from address | The address from which emails sent by the BI Server originate. |
| User ID | ID used to connect to the email server for sending email. |
| SMTP Host | Address of your SMTP email server for sending email. |
| SMTP Port | Port of your SMTP email server; Usually this value is 25, for Gmail the value is 587. |
| SMTP Protocol | Transport for accessing the email server. Usually this is SMTP. For Gmail this is SMTP. |

| Setting | Description |
|---|---|
| Use SSL | Enable if the email server requires an SSL connection. For Gmail this value must be enabled. |
| Enable Debugging | Enable to output email-related debugging information from the JavaMail API to the BI server log file. |

4. Click **Submit**.
   A success message appears.

## Specifying Your Publisher Password

The publisher password enables client tools such as the Pentaho Report Designer and the Pentaho Metadata Editor to publish content on the BI Server.

To specify your publisher password...

1. In the Pentaho Enterprise Console home page, click **Configuration**, then click the **General** tab.
2. In the **Publisher** group box, enter the old password in the **Old Password** text box; enter the new password in the **New Password** text box.
3. Click **Submit** to update the password.
   A success message appears.

## Specifying Global and Session Variables

To specify global and session variables...

1. In the Pentaho Enterprise Console home page, click **Configuration**.
2. Click the **General** tab.
3. In the **System Actions** group box, there are global and session action sequences that run once at Pentaho startup and once per user session, respectively. Use the controls to add, delete, and edit existing global and session action sequences. The table below provides description associated with action sequences:

| Item | Description |
|---|---|
| Action Sequence Path | Solution path to the action sequence |
| Scope | Either Global or Session |
| Type | Either Servlet or Portlet |

## Changing the Location of the Server Log File

If you used the install wizard to install the Pentaho BI Suite, the pentaho.log is written to the C:\WINDOWS\system32 directory. To change the location of the pentaho.log file, you must edit the log4j.xml available at `/pentaho/server/ biserver-ee/tomcat/webapps/pentaho/WEB-INF/classes/`.

Modify the location as shown in the sample below using the appropriate path to your installation of the Pentaho BI Suite.

```
<param name="File" value="C:\Program Files\pentaho\server\biserver-ee\logs\pentaho.log"/
>
<param name="Append" value="true"/>
```

## Changing the Port on which Tomcat is Running

Follow the instructions below to change the port on which Tomcat is running:

1. Load `...\biserver-ee\tomcat\conf\server.xml`.

**2.** Locate the following line:

```
<Connector URIEncoding="UTF-8" port="8080"...
```

**3.** Change the port value as needed.

**4.** Save the **server.xml** file and restart Tomcat.

After changing the port number for Tomcat, you must modify the BI Server **fully-qualified-server-url** to use the new port number. You can do this either in the Pentaho Enterprise Console or directly in the BI Server web.xml file.

In the `...pentaho\WEB-INF\web.xml`, change the port number here:

```
<context-param>
            <param-name>fully-qualified-server-url</param-name>
            <param-value>http://localhost:8080/pentaho/</param-value>
</context-param>
```

In the Pentaho Enterprise Console, go to **Configuration** -> **Web Settings** and type the correct path in the **Base URL** text field.

# Configuring and Monitoring the Content Repository (Hibernate)

The content (or "managed") repository refers to the database that stores solution content, runtime content (including audit log data), and schedule information for your BI Server. It is critical that this repository be available and working correctly for your BI server to function properly.

## Checking the Status of the Managed Repository

To determine if your managed repository is responding to query requests...

1. In the **Pentaho Enterprise Console,** go to the **Status** page.
2. Under the **Repository** tab, in the **Information** group box, you see an item labeled "**Can the Pentaho Enterprise Console connect to the repository?**."
   If the value for this line is **true**, your repository is up and receiving query requests.

## Configuring the Managed Repository

You can change the server's repository configuration in the Pentaho Enterprise Console **Configuration** page. This configuration enables the server to communicate with the Managed Repository, using *Hibernate*. For example, you may want to change the configuration so that the server can access information in a production database.
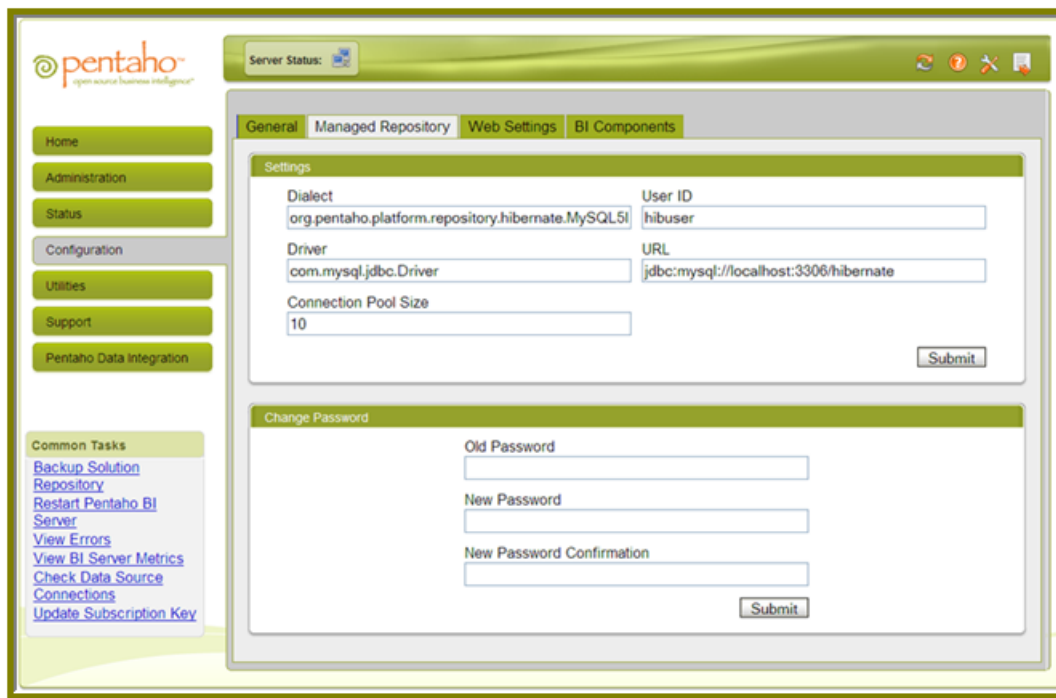


Changing the user name and/or password used to authenticate with the managed repository is likely the most frequent change to this configuration. You can, however, change other connection information, such as the JDBC driver class name, the Hibernate dialect to use for your repository, and the connection URL to the database.

⚠️ **Caution:** Do not make changes to the Managed Repository configuration unless you are certain you know what you are doing. Improper changes to this configuration can cause your server to stop working.

| To change this configuration setting... | Follow these instructions... |
|---|---|
| **Driver** | Enter a new driver class name in the text box and click **Submit**. |
| **User Id** | Enter a new user ID in the text box and click **Submit**. |
| **Connection Pool Size** | Enter a new connection pool size in the text box and click **Submit**. |
| **Dialect** | Enter a new dialect string in the text box and click **Submit**. |

| To change this configuration setting... | Follow these instructions... |
|---|---|
| URL | Enter a new connection URL in the text box and click **Submit**. |
| Change Password | Enter the old database repository password in the **Old Password** text box. Enter the new database repository password in the **New Password** text box. Click **Submit**. |

## Hibernate Dialects

Below is a list of dialects for each database type associated with the Managed Repository:

| Database Type | Dialect |
|---|---|
| HSQLDB ("Hypersonic" — not for production use) | org.hibernate.dialect.HSQLDialect |
| MySQL | org.hibernate.dialect.MySQL5InnoDBDialect |
| Oracle 10g | org.hibernate.dialect.Oracle9Dialect |
| PostgreSQL | org.hibernate.dialect.PostgreSQLDialect |

# Cleanup and Audit Reports

The **Services** page associated with the **Administration** menu item provides you with the ability to manage system-level services. Many of the options on this page are associated with managing services related to the solution and content repositories; for example, clearing old data from the content repository and restoring the solution repository. Also included are tools that allow you to audit usage of data in the repository; for example, auditing helps you determine what type of repository-related data (such as reports) is generating the highest interest among users. The table below provides you with a short description of each option on the Services page:



| Option | Description |
|---|---|
| **Schedule** | Schedules the daily removal of files created in the content repository located at `/pentaho-solutions/system/content/`, which are over 180 days old. To change, the number of days, edit the solution file **clean_repository.xaction** located in /pentaho-solution/admin. To change the recurrence, edit the solution file, **schedule-clean.xaction** located in /pentaho-solution/admin |
| **Execute** | Removes files created in the content repository located at `/pentaho-solutions/system/content/` that are over 180 days old. To change, the number of days, edit the solution file, **clean_repository.xaction** located in `/pentaho-solutions/admin/`. |
| **Refresh** | Synchronizes the RDBMS-based solution repository with the file system when solution files are manually added or edited |
| **Backup** | Backs up ACLs to pentaho.xml, zips pentaho-solutions and places it in the configured backup directory (see Note) |
| **Version Control** | Updates the solution files on the local file system from a version control system (SVN/CVS). The link executes a script called **versionControl.bat** or **versionControl.sh** found in the solution repository. The script commands get an update from the version control system into the local file system. |
| **Reset Permissions** | Deletes all the solution files and their permissions from the RDBMS-based solution repository and copies the files with default permissions from the master solution repository on the local file system. |
| **Restore** | Restores system from a backup list (restores ACLs and/or solutions) (see Note) |

| Option | Description |
|---|---|
| **System Settings** | Reloads all the configurations files found in pentaho-solutions/System. |
| **Global Variables** | Updates global variables by re-executing registered solution files. |
| **Metadata Models** | Refreshes the Metadata cache when models are added, edited, or deleted in the solution repository |
| **Mondrian Cache** | Clears the Mondrian schema and data cache. |
| **Initialize from File** | Loads the audit log file into a database used for Audit Reports and Ad-hoc Audit Reports when using the file-based solution repository. |
| **Update from Table** | Loads the audit log table into a database used for Audit Reports and Ad-hoc Audit Reports when using the RDBMS-based solution. repository, |
| **View Reports** | Provides a list of over 20 pre-built reports for analyzing report usage, user access, and report processing time. |
| **View Ad Hoc Reports** | Provides an interface for administrators to create their own queries of the audit information. |

> **Note:** The **Backup** and **Restore** options do not commit until progress is 100%; therefore, canceling is non-destructive. Resetting ACLs returns to the last restore (or default).

## More About Audit Reports

Audit reports help you understand and optimize your business intelligence applications. Audit reports provide immediate insight into user activity, system performance, most popular reports, and more. Audit reports can also be useful for tracking compliance and ensuring that resources are used according to company policy. Beyond that, understanding system performance and most popular reports allows you to progressively tune your deployment, and ensure that business users have the access to the most critical information necessary for doing their jobs.

In the sample below, the administrator has clicked **View Reports** under **Services** in the Pentaho Enterprise Console to open the Audit Reports feature in the Pentaho User Console:



The administrator then clicks first arrow next to **Most Frequently Used Content** to display the .xaction (solution) files that are most frequently being accessed by users in the last 24 hours:
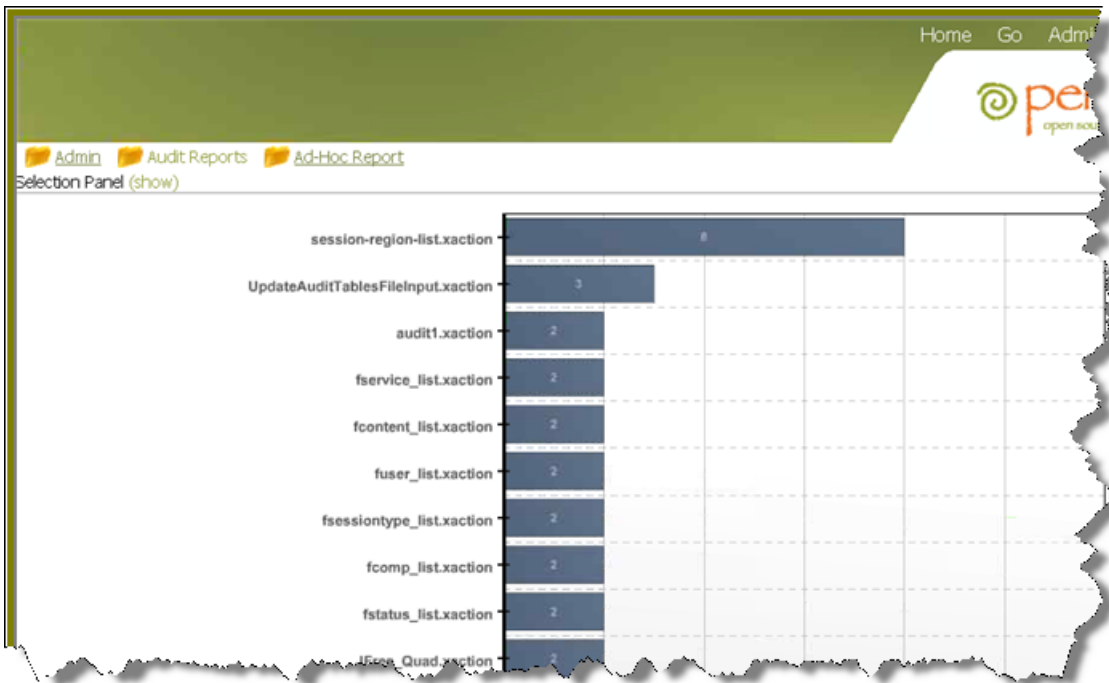
The administrator can continue to drill down to obtain usage information for the previous week, month-to-date, all time, and more.

**SQL-Based Versus File-Based Audit Logging**

Out of the box, the BI Server is configured to log all audit entries to a SQL table. SQL-based logging offers the advantage of incremental updates to the audit reporting tables. The BI Server can also be configured to log audit entries to a text file. The file-based audit logger streams all audit information to a text file.

The type of audit logging you select depends on the requirements of your environment. High-use production environments result in a significant number of SQL inserts that could impact server performance. In this case, you may choose to switch to file-based audit logging. See *Switching to File-Based Audit Logging* for instructions.

# Error Logging, Performance Monitoring, and Troubleshooting

This section provides you with instructions for diagnosing BI server errors and information about how to use the server performance monitoring tools.

## Viewing Errors

To view the list of BI Server warnings and errors...

1. In the Pentaho Enterprise Console home page, click **Status**.
2. Click on the **Errors** tab to display all the errors that the Pentaho Enterprise Console has identified.
3. To resolve the individual errors, go to the specific location in the console from which the warning or error originated.

## Viewing Environment Settings

Environment settings provide you with detailed information about about your "environment" such as the operating system you are using, the Java version you are using, the address of your Tomcat server, and more.

To view the Environment Settings...

1. In the Pentaho Enterprise Console home page, click **Status**.
2. Click on the **Environment** tab to display the Java Virtual Machine (JVM) system settings of the application server on which the BI Server is running

> **Note:** If the application server is not configured or not online these values are not available.

## Viewing Security Settings

Many of the security settings are distributed across parts of the Pentaho Enterprise Console:

> **Note:** The *Pentaho BI Platform Security Guide* has detailed instructions for configuring BI Server and Pentaho Enterprise Console security.

| Setting | Description |
|---|---|
| **Access Control** | To view and modify the Access Control settings, in the Pentaho Enterprise Console home page, click **Configuration**, then click on the **General** tab. The Access Control group box allows you to customize **Access Control Lists** (ACLs) and other low level access control settings. |
| **Authentication** | To view and modify the **Authentication** setting, in the Pentaho Enterprise Console home page, click **Configuration**, then click on the **Web Settings** tab. The current authentication options include memory-based, RDBMS-based, and LDAP-based. |
| **LDAP Configuration** | To view and modify the LDAP configuration, in the Pentaho Enterprise Console home page, click **Utilities**, then click on the **LDAP Configuration** tab. Refer to the *Pentaho BI Platform Security Guide* for additional documentation about the LDAP Configuration. |

### LDAP

The Pentaho Enterprise Console has an LDAP section in the **Utilities** category. The fields in this screen should be relatively intuitive, but are explained below in more detail in case there are options that you are not familiar with:

| Property Group Name | Description |
|---|---|
| **Directory Connection** | Properties used for connecting to an LDAP directory (such as the host name). |
| **User Search** | Properties that describe a query that gets a user's record using his or her user name. When users attempt to log on they provide a user name and password. The BI Server uses the user name given at logon, along with the properties in this group, to get the user's record. |
| **Authorities Search** | Properties that describe a query that gets all roles that should be recognized by the BI Server. The BI Server needs a complete list of roles for the purpose of exposing them in an interface used to assign permissions to action sequences. (This is the "Permissions" interface.) |
| **Populator** | Properties that describe a query that gets a user's roles using his/her user name or Distinguished Name (DN). Because there is no standard for where to store roles in a directory, roles could be stored anywhere in the directory tree. To support this inconsistency, the BI Server provides a separate search. |

# Monitoring Performance

The Metrics page associated with the **Status** menu item provides key performance metrics for your BI Server. The metrics provided give you insight into the latest action sequence execution activity on your server, the latest user activity, and action sequence execution analysis over a specified period of time.

**Performance Metrics Values**

The atomic performance values are all based on the **Show for last (minutes)** interval (with the exception of the Logged In Users metric). Each metric is calculated relative to this interval, which is the number of minutes from the time the metrics were last refreshed. For example, if your interval is set to 15 minutes, the Average Duration metric displays the average execution time for all action sequences executed in the last 15 minutes ("last" meaning the time the metrics were last refreshed). The performance metrics are calculated using data gathered from the BI Server Enterprise Edition RDBMS audit logging engine.
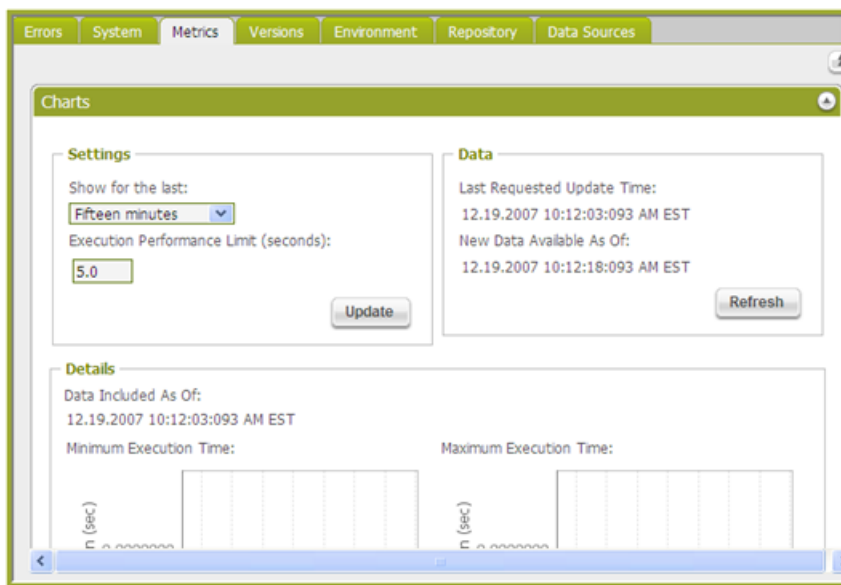


**Note:** By default, the BI Server is configured to log audit information to an RDBMS. If you are not using the default configuration, performance metrics may not be available.

The table below provides a description of what each metric measures:

| Metric | Description |
|---|---|
| **Average Duration** | The average execution time for all action sequences executed in the last metrics interval. |
| **Maximum Duration** | The maximum execution time for all action sequences executed in the last metrics interval. |
| **Minimum Duration** | The minimum execution time for all action sequences executed in the last metrics interval. |
| **Number of Executions** | The number of action sequences executed in the last metrics interval. |
| **(Users) Current Active/Logged in** | The number of users who have completed execution of at least one action sequence in the last metrics interval. Note that this user may no longer be logged on, or have an active session. The only thing this metric indicates is that the user performed some activity within the last interval. |
| **Number of Logged In Users** | This metric displays the current number of authenticated users that are logged on. Users are added to the count of logged on users when they log onto the server, and are removed when their session times out or they manually log off. This metric does not include users who anonymously access the server. This value is refreshed only when the performance metrics are refreshed manually. |

## Changing the Show for Last Interval

To change the **Show for Last interval** to a more relevant value for your environment...

1. Enter a new value in the **Show for Last** text box. This value can be any decimal number and represents the interval in **minutes**.
2. Click **Update**.
   Changing the Show for Last value updates the performance metrics.

## Refreshing the Metrics Values

Metrics values must be refreshed manually. There is no automatic update of the metrics values, except at start up of the Pentaho Enterprise Console. To refresh metric values, click **Submit**. Each metric is updated, and the **Last Updated** value is updated to reflect the latest refresh date and time.



**Note:** Refreshing the performance metrics values does not refresh the performance charts. See *Refreshing the Performance Charts* for more information on updating the performance charts.

## Preventing Timeout Transaction Errors Associated with Scheduled Reports

When you create a schedule to which you add multiple reports, it's possible to add so many reports that the Subscription (Public Schedule) crashes. The crash occurs because the Hibernate Managed Subscription is unable to complete each transaction in under 50 seconds. The result includes state exceptions, and other problems, on

subsequent runs that eventually lead to a Java heap space out-of-memory error. The issue occurs when running MySQL.

Follow the instructions below correct the problem:

1. Stop the BI server and MySQL, consecutively.
2. Using a text editor, open the MySQL **my.ini** file (in Windows) or the **my.cnf** file (in Linux/MAC).
3. Locate the following line: **innodb_lock_wait_timeout = 50**

   If you don't see this parameter, add it to the file.
4. Change the timeout from 50 to a higher number that can handle the load for a single (highly loaded) public schedule and save the file.

   > **Note:**
   >
   > You may need to adjust the timeout value to find the right threshold. Setting it way too high is not recommended because in instances where a transaction cannot complete because of an error, the thread hangs for the timeout length.

5. Restart the BI server and MySQL, consecutively.

## Using Performance Charts

The performance charts are all based on the **Show for the last** period selected. The time periods available to choose from are:

- Fifteen minutes
- Thirty minutes
- One hour
- Twelve hours
- 24 hours
- Seven days
- Thirty days
- Ninety days

Each chart plots data available from the BI Server audit logging engine. Data is selected that falls within the selected time period, and is then aggregated and/or calculated according to that chart's purpose. For example, if you select the "One hour" time period, then the charts display the trend for action sequence executions over the last hour ("last" meaning the time the charts were last refreshed).

| Chart Type | Description |
|---|---|
| **Minimum Execution Time Chart** | Displays the minimum amount of time it took for action sequences to execute over the Analysis Time Period. |
| **Maximum Execution Time Chart** | Displays the maximum amount of time it took for action sequences to execute over the Analysis Time Period. Average Execution Time Chart |
| **Number of Executions Chart** | Displays how many action sequences executed in intervals over the Analysis Time Period. Execution Performance Chart |
| **Execution Performance Chart** | Displays the percentage of action sequences (on the Y axis) that executed within a given number of seconds (the x axis) over the Analysis Time Period. This is a cumulative chart, so all action sequences that took less than one second also took less than 2 seconds, and less than 3 seconds, and so on. |
| **Execution Performance Limit Deviation Chart** | Displays the percentage of action sequences that executed successfully within a selected Execution Performance Limit. This limit, expressed in seconds, is configurable. See *Changing the Execution Performance Limit for the Execution Performance Limit Deviation Chart* |

### Changing the Chart Analysis Time Period

To change the analysis time period to a more suitable value that makes sense in your environment...

1. Select a new value in the **Show for the last** selection list.
2. Click **Submit**.

## Changing the Execution Performance Limit for the Execution Performance Limit Deviation Chart

Once you have analyzed your execution data, you will have a better idea of what your acceptable performance limit for action sequence execution is. At that point, you may want to change the **Execution Performance Limit**.



1. Enter a new value in the **Execution Performance Limit** (seconds) text box. This value can be any decimal number, and represents the limit in seconds.
2. Click **Submit**.
   Changing the Execution Performance Limit updates the performance charts.

## Refreshing the Performance Charts

Currently, there is no automatic update of the performance charts, except at start up of the Pentaho Enterprise Console. To manually update the charts, under **Data**, click **Refresh**.



All charts are updated, and the **Last Requested Update Time** is updated to reflect the latest refresh date and time. The **Data Included As Of** date is updated. The Data Included As Of date value represents the date and time limit for data included in the chart. This differs from the Last Requested Update Time due to calculating the chart data intervals over multiple updates to the charts and attempting to keep the data displayed consistently from one update to the next. And last, the **Data Available As Of** date value is updated, which reflects the next date and time new data will be available to the charts. If you attempt to update the charts before this date and time, the charts will not update, as new data is not yet available.

> **Note:** This availability is based on the analysis time period. By changing the analysis time period, you essentially reset these dates, and the charts update immediately, regardless of the value of the **Data Available As Of** date value.

> **Note:** Refreshing the performance charts does not refresh the performance metrics values. See *Refreshing the Metrics Values*for more information on updating the performance metrics values.

## Estimating the Size of the Audit Log

Turning on audit logging means that the size of your solution repository will grow at a faster rate. Some DBAs may need to accommodate for this accelerated growth, so this section explains how to calculate audit log growth.

First of all, the size of the audit log does not depend on the logging level used.

The average length of an audit entry is 220 characters. Each of the following events is logged:

- Session start
- Action sequence execution start
- BI component execution start
- BI component execution end
- Action sequence execution end

An action sequence with four actions defined in it will generate about 2200 characters in the audit log. 2200 = 220 x 10. The 10 comes from 2 (for the xaction start and end) + (2 * 4) (for each component).

If the DB-based audit option is used, the raw DB-based audit entries will be the same size or slightly smaller.

The raw DB tables or log files are transformed via ETL into the audit reporting tables. These tables are smaller than the raw tables as the start and end rows are collapsed into single rows, so these tables will be about half the size of the raw tables.

The raw log files or tables do not have to be kept once they have been processed into the audit reporting tables.

Given these variables:

- **U** = average number of users per 24-hour period
- **R** = average number of reports (xactions) run per user per day
- **C** = average number of components per xaction

Approx size of raw logs (file or DB) = U * R * ( 440 + C * 440 )

Approx size of audit reporting tables = U * R * ( 220 + C * 220 )

A slightly simpler version of the calcs:

Approx size of raw logs (file or DB) = U * R * (C+1) * 440

Approx size of audit reporting tables = U * R * (C+1) * 220

Another way to look at it is an xaction with 3 components will add, per execution, about 1.7k of raw log and 0.86k of audit reporting data.

### Switching to File-Based Audit Logging

In high-use environments, performance issues may occur due to a sizeable number of SQL inserts associated with audit logging. In this instance, you can switch to file-based audit logging.

1. Open the file, **pentahoObjects.spring.xml**, located at `...\pentaho-solutions\system`.
2. Locate the following line:

```
<bean id="IAuditEntry"
 class="org.pentaho.platform.engine.services.audit.AuditSQLEntry"
scope="singleton" />
```

3. Edit the line as follows:

```
<bean id="IAuditEntry"
 class="org.pentaho.platform.engine.services.audit.AuditFileEntry"
                       scope="singleton" />
```

4. Save the file and restart the BI Server.

### Removing Audit Logging

Follow the instructions below if you do not want to use audit logging. You may want to disable audit logging when you are in a test or development environment. Disabling audit logging improves performance; however, make sure you have no need to evaluate data contained in audit logs.

1. Go to `...\pentaho-solutions\system\`.
2. Open the **pentahoObjects.spring.xml** file
3. Locate the line: `<bean id="IAuditEntry" class="org.pentaho.platform.engine.services.audit.AuditSQLEntry" scope="singleton" />`
4. Replace the line above with this line: `<bean id="IAuditEntry" class="org.pentaho.platform.engine.core.audit.NullAuditEntry" scope="singleton" />`
5. Save the file and restart the BI Server.

## Defining Result Row Limit and Timeout

When a Web-based Ad Hoc query in the Pentaho User Console returns an unusually large number of rows, this may impact server performance. To limit the number of rows returned by a query and to set up a timeout, you must create two custom properties, **max_rows** and **timeout**, in the Pentaho Metadata Editor.

👉 **Note:** The values you define for the row number limit (max-rows) and timeout properties are passed to the JDBC driver.
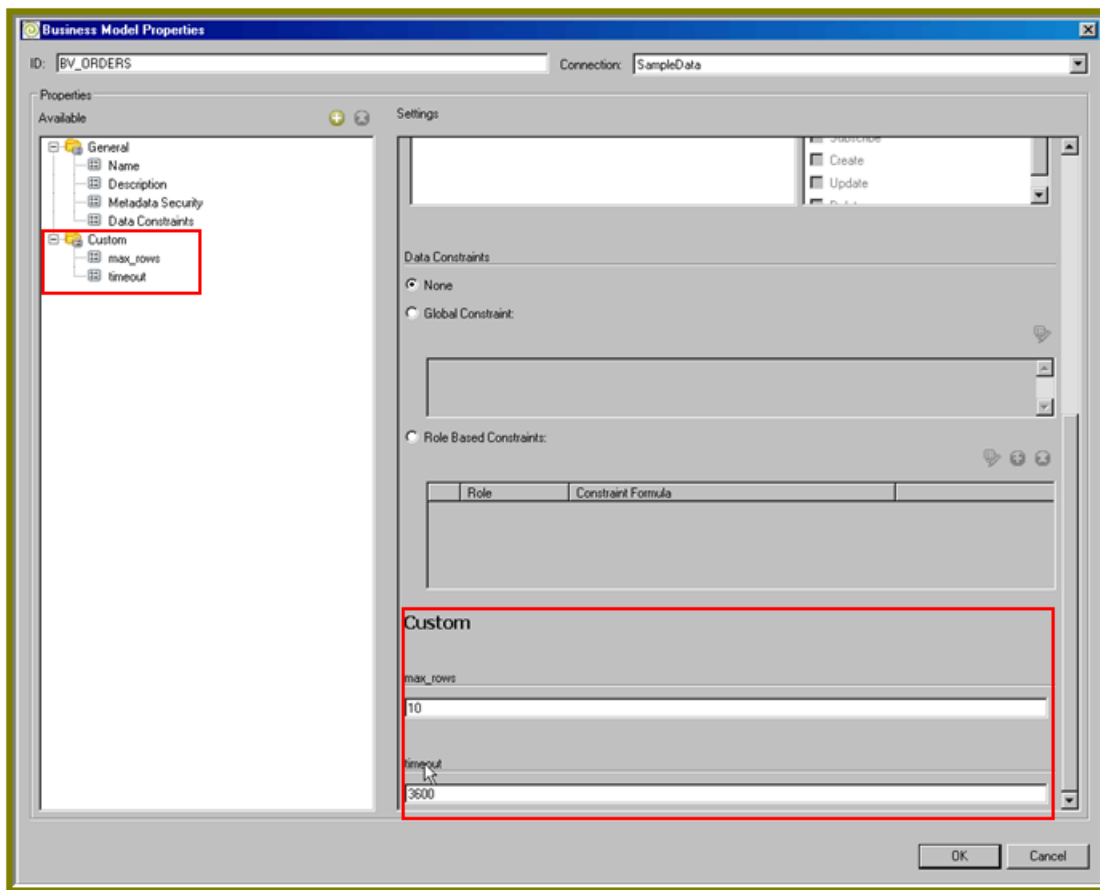


To define max rows and timeout...

1. In the Pentaho Metadata Editor, expand the **Business Model** node and select **Orders**.
2. Right-click Orders and choose **Edit**.
   The **Business Model Properties** page displays a list of properties that were previously defined.
3. In the Business Model Properties page, click ⊕ (Add).
   The Add New Property page dialog box appears.
4. Enable **Add a custom property**.

5. In the **ID** text box, type **max_rows**.

> **Important:** The ID is case-sensitive and must be typed exactly as shown.

6. Click the down-arrow in the **Type** field and choose, **Numeric**.
   The **Business Model Properties** page appears. The **max_rows** property is listed under **Custom** in the navigation tree.
7. In the right pane, under **Custom**, enter a value for your **max_rows** property. For example, if you enter "3000" as your value, the number of rows allowed to display in a query result is constrained to 3,000.
8. Repeat steps 3 through 6 to for the **timeout** custom property.
9. In the right pane, under **Custom**, enter a value for your **timeout** property. The timeout property requires a numeric value defined in number of seconds. For example, if you enter, 3600, the limit for query results is one minute.
10. Click **OK** in the Business Model Properties page to save your newly created properties.

## Column Metadata Security Restrictions On Columns With Inner Joins

You may experience malformed SQL query results when creating ad hoc reports if your metadata model contains security restrictions on columns that are joined to other columns. This is not a supported action in Pentaho Metadata's security infrastructure. In the BI Suite version 3.5.2-GA and later, an error message will appear that explains this limitation. For all versions prior to 3.5.2, the BI Server will attempt to generate an improperly formed SQL query, which can have unpredictable results that are difficult to troubleshoot.

## Disabling Server and Session-Related Timeouts in the Pentaho User Console

Follow the instructions below to disable server and session timeouts associated with the **Pentaho User Console**.

> **Important:** These instructions are applicable when you are in a *test* environment. Once you go live, it is recommended that you set your timeouts to five or ten minutes so that sensitive BI Server-related data can be protected.

1. Open the file, **server.xml**, located under …\biserver-ee\tomcat\conf.
2. Find the **connectionTimeout="20000"** parameter and change its value to zero ("**0**").
3. Open the file **web.xml**, located under ...  \bi-server_ee\tomcat\webapps\penaho\WEB-INF\web.xml.
4. Find the **session-timeout** parameter and changes its value to zero ("**0**")
5. Save the file and refresh the user console.

## Refreshing the Reporting Data Cache

By default, reports created with Report Designer and published to the BI Server will have their data sets cached to improve performance. If you are working with a volatile data source and will be regenerating reports frequently, you may need to manually flush the report cache, which will force the Reporting engine to recalculate the result sets for all reports.

To do this, log into the Pentaho User Console as an administrator user, then go to the **Tools** menu, select the **Refresh** sub-menu, and click the **Reporting Data Cache** item.

## Troubleshooting

This section contains known problems and solutions relating to the procedures covered in this guide.

## Case-Insensitivity for Usernames, Passwords, and Filenames

This problem can manifest in two different known ways. It can show as case-insensitivity as mentioned above, or it can show up when you change the case of an action sequence file name.

The source of the problem is the collation method and character set for your MySQL server. Selecting the wrong configuration will cause all usernames, passwords, and content files stored in the database to become identical in terms of characters without regard for case; this can cause name collisions. SUZY, Suzy, and suzy will all be the same

username as far as the BI Server will be able to tell. To change this, you must set your database or table collation method to **latin1_general_cs**. You can do this by editing the database scripts mentioned below, or by modifying your MySQL server configuration.

## HTTP 500 or "Unable to connect to BI Server" Errors When Trying to Access Enterprise Console

If you have trouble controlling the BI Server through the Pentaho Enterprise Console, it is likely because you've changed the BI Server's IP address or hostname (the **fully-qualified-server-url** node in `web.xml`) from the default 127.0.0.1 to something else without also applying this change to the Proxy Trusting Filter in the `web.xml` file. See *Configuring the Proxy Trusting Filter* on page 9 for instructions on how to fix this.

# Getting Support

The Support menu item contains four group boxes that provide you with assistance when a BI server problem arises:

- The Export Settings group box allows you to create an XML file that contains your BI server settings and configuration to send to the Pentaho support team. Support may request that you send this file when they need to diagnose an issue. To export your configuration and settings to an XML file, click Export to **XML**.
- The **Links** group box contains links to helpful articles that may help you address a BI server-related issue.
- The **FAQ** group box contains questions and answers to common issues associated with your BI server.
- The **How Tos** group box contains updates for issues you submit to support.

# Appendix: Working From the Command Line Interface

Though the Pentaho Enterprise Console is the quickest, easiest, and most comprehensive way to manage PDI and/or the BI Server, some Pentaho customers may be in environments where it is difficult or impossible to deploy or use the console. This appendix lists alternative instructions for command line interface (CLI) configuration.

## Installing an Enterprise Edition Key on Linux (CLI)

To install a Pentaho Enterprise Edition Key from the command line interface, follow the below instructions.

☞ **Note:** Do not open your LIC files with a text editor; they are binary files, and will become corrupt if they are saved as ASCII.

1. Navigate to the `/pentaho/server/enterprise-console/license-installer/` directory, or the `/license-installer/` directory that was part of the archive package you downloaded.
2. Run the **install_license.sh** script with the **install** switch and the location and name of your license file as a parameter. You can specify multiple files, separated by spaces, if you have more than one license key to install.

    ☞ **Note:** Be sure to use backslashes to escape any spaces in the path or file name.

    ```
    install_license.sh install /home/pgibbons/downloads/Pentaho\ BI\ Platform\ Enterprise\ Edition.lic
    ```

Upon completing this task, you should see a message that says, "The license has been successfully processed. Thank you."

## Installing an Enterprise Edition Key on Windows (CLI)

To install a Pentaho Enterprise Edition Key from the command line interface, follow the below instructions.

☞ **Note:** Do not open your LIC files with a text editor; they are binary files, and will become corrupt if they are saved as ASCII.

1. Navigate to the `\pentaho\server\enterprise-console\license-installer\` directory, or the `\license-installer\` directory that was part of the archive package you downloaded.
2. Run the **install_license.bat** script with the **install** switch and the location and name of your license file as a parameter.

    ```
    install_license.bat install "C:\Users\pgibbons\Downloads\Pentaho BI Platform Enterprise Edition.lic"
    ```

Upon completing this task, you should see a message that says, "The license has been successfully processed. Thank you."

## Managing Enterprise Edition Keys (CLI)

To list or remove Pentaho Enterprise Edition keys in the command line interface, follow the below instructions.

1. Navigate to the `/pentaho/server/biserver-ee/enterprise-console/license-installer/` directory.
2. Run the **install_license.bat** (on Windows) or **install_license.sh** (on Linux) script with the **display** switch.

    ```
    install_license.sh display
    ```

    If you have installed any Enterprise Edition keys, a list of them will appear, along with the products they cover and the term of the support contract.
3. To remove a license, run the **install_license** script with the **uninstall** switch.

    ```
    install_license.bat uninstall
    ```

A list of installed licenses will appear, followed by a prompt for the license ID you would like to remove. If you press Enter at this prompt, it will exit without taking any action.

4. Type in the license ID number that you want to remove, then press Enter.

After removing a key, if you had more than one installed, the list will regenerate and the prompt will reappear. You can choose to remove another Enterprise Edition key, or you can press Enter to exit the script.

> **Note:** If you would prefer not to be prompted for confirmation, or if you intend to call this program as part of a script, use the **-q** switch to suppress prompting.

## Resetting the Default Access Control Lists (CLI)

Follow the below process to reset the default ACLs from the command line interface.

1. Stop the Pentaho BI Platform service or server.

> **Note:** The solution database server must remain running for this procedure.

```
sh /usr/local/pentaho/server/biserver-ee/stop-pentaho.sh
```

2. Execute the following SQL statements using your database's command line utility:

   **mysql**, **psql**, and **sqlplus** are the preferred utilities for MySQL, PostgreSQL, and Oracle, respectively.

```
DROP TABLE PRO_ACLS_LIST;
DROP TABLE PRO_FILES;
```

3. Start the Pentaho BI Platform service or server.

```
sh /usr/local/pentaho/server/biserver-ee/start-pentaho.sh
```

The dropped tables have been recreated with default values.