pentaho

Version 5.4

Set Up BA Server and Tools

Copyright Page

Help and Support Resources

To view the most up-to-date help content, visit https://help.pentaho.com.

If you do not find answers to your questions here, 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, visit http://www.pentaho.com/training.

Liability Limits and Warranty Disclaimer

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

The trademarks, logos, and service marks ("Marks") displayed on this website are the property of Pentaho Corporation or third party owners of such Marks. You are not permitted to use, copy, or imitate the Mark, in whole or in part, without the prior written consent of Pentaho Corporation or such third party. Trademarks of Pentaho Corporation include, but are not limited, to "Pentaho", its products, services and the Pentaho logo.

Trademarked names may appear throughout this website. Rather than list the names and entities that own the trademarks or inserting a trademark symbol with each mention of the trademarked name, Pentaho Corporation 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.

Third-Party Open Source Software

For a listing of open source software used by each Pentaho component, navigate to the folder that contains the Pentaho component. Within that folder, locate a folder named licenses. The licenses folder contains HTML.files that list the names of open source software, their licenses, and required attributions.

Contact Us

Global Headquarters Pentaho Corporation Citadel International, Suite 460

5950 Hazeltine National Drive Orlando, FL 32822

Phone: +1 407 812-OPEN (6736)

Fax: +1 407 517-4575

http://www.pentaho.com

Sales Inquiries: sales@pentaho.com

Introduction

In this section, you do the minimum configuration tasks for the Pentaho Business Analytics (BA) Server, User Console, and design tools, so you can get started creating data analytics, reports, and dashboards.

Prerequisites

Before you begin, you must have installed Pentaho software. If you chose to install the DI server and its design tool, there is a separate section for configuring them.

Expertise

The topics in this section are written for IT administrators who know where data is stored, how to connect to it, details about the computing environment, and how to use the command line to issue commands for Microsoft Windows or Linux.

Tools

We provide a web application, the User Console, which you use to perform most configuration tasks.

Login Credentials

All the tasks that use the User Console, **Administration** page, require that you log on to the User Console with the Pentaho administrator user name and password.

Configure the BA Server

Before you can create Web-based data analysis, reports, and dashboards, you need to perform configuration tasks for the BA Server through the User Console. Just follow the *Guide Post* graphics to ensure you complete the entire process. It should take you approximately 3 hours or less.

Start and Stop BA Server

To work with the User Console, the BA repository and the BA server have to be running. Yet, there may be times when you need to manage these components individually, stopping them to perform system maintenance and restarting them after you are done. As part of the Installation Wizard and Archive Installation, we provide different ways to help you start and stop the components, depending on the operating system you use. If you installed Pentaho software manually, use the information in this section as a guide to create a custom way to start and stop these components.
For Microsoft Windows, these are your options for starting and stopping the components.

- For the Installation Wizard
- For the Archive Installation

For Linux, these are your options for starting and stopping the components.

- For the Installation Wizard
- For the Archive Installation
- For the Installation Wizard and Archive Installation—starting on boot

When you are done, please go on to the next stop on the *Guide Post* graphic.

Windows Pentaho Installation Wizard

When you ran the Installation Wizard on Windows, the BA Server is deployed in an Apache Tomcat application server. You can manage the BA Server by clicking **Start** > **All Programs** > **Pentaho Enterprise Edition** > **Server Management** and then select one of these menu items.

- Start BA Server
- Stop BA Server

The wizard also registered the BA Server, as well as the BA repository, as services. These services are set to run automatically, enabling them to start and stop when the computer running them boots or shuts down. You can use the Windows Services applet found in the Control Panel to start and stop the Pentaho servers and the BA repository.

1. Click **Start** > **Control Panel** > **Administrative Tools** > **Services**.
2. In the **Services** window, right-click on one of these services in the list and take the appropriate action.
    - Pentaho BA Server
    - Pentaho BA Repository

Windows Pentaho Archive Installation

If you used the Archive Installation, we provide individual control scripts to start and stop the BA server and BA repository. Here is where you can find the individual control scripts.

BA Repository

- The Archive Installation enables you to install either PostgreSQL, MySQL, MS SQL Server, or Oracle as the repository. Consult the documentation for the RDBMS you selected for information about starting and stopping.

  The BA repository must be started before the BA Server.

BA Server

- `/pentaho/server/biserver-ee/start-pentaho.bat and stop-pentaho.bat`

Linux Pentaho Installation Wizard

When you ran the Installation Wizard on Linux, the BA Server is deployed in an included Apache Tomcat application server. You can control the Tomcat server using the start and stop scripts that come with the Pentaho installation. This script is also used as an easy way to start and stop the BA Server and the PostgreSQL repository.

You can find this script at `/pentaho/ctlscript.sh`.

Here is a list of the script arguments and services you can use.

Arguments

- `start`
- `stop`
- `restart`
- `status`
- `help`

Services

- `baserver`
- `postgressql`

```
./ctlscript.sh start baserver
```

```
./ctlscript.sh status postgressql
```

```
./ctlscript.sh help
```

Linux Pentaho Archive Installation

If you used the Archive Installation, Pentaho provides individual control scripts to start and stop the Tomcat application server, BA Server, and BA repository. You can find these scripts at `/pentaho/server/biserver-ee`.

BA Repository

- The Archive Installation enables you to install either PostgreSQL, MySQL, or Oracle as the BA repository. Consult the documentation for the RDBMS you selected for information about starting and stopping.

  The BA repository must be started before the BA Server.

BA Server

```
./start-pentaho.sh
```

```
./stop-pentaho.sh
```

Linux Pentaho Wizard or Archive Installation—Starting on Boot

The Installation Wizard and the Archive Installation do not provide a way to start the BA repository and BA Server automatically on boot. They also do not stop automatically at shutdown. Here are examples of how you might approach creating a script to start at boot and stop at shutdown.

This procedure assumes that you are running the BA Server under the `pentaho` local user account. If you are using a different account to start these services, substitute it in the script in step 2. This script also assumes you are using the PostgreSQL repository. Where `postgresql` appears in this script, change it to reflect the RDBMS you are using as a repository, either MySQL or Oracle. This script was tested on Red Hat Enterprise Linux. You may have to modify the details of the script if you use a different distribution of Linux or other Unix-like operating system, different shells, or different `init` systems.

1. With root permissions, create a file in `/etc/init.d/` named `pentaho`.
2. Using a text editor, copy the following content into the new pentaho script. If running the solution repository on the same machine as the server, change `postgresql` to the name of the `init` script for your database. If running the solution repository on the a remote computer, remove `postgresql` entirely. You may also have to adjust the paths to the BA Server scripts to match your situation.

   ```
   #!/bin/sh
   ### BEGIN INIT INFO
   # Provides: start-pentaho stop-pentaho
   # Required-Start: networking postgresql
   # Required-Stop: postgresql
   # Default-Start: 2 3 4 5
   # Default-Stop: 0 1 6
   # Description: Pentaho BA Server
   ```

```
### END INIT INFO

case "$1" in
"start")
su - pentaho -c "/home/pentaho/pentaho/server/biserver-ee/start-pentaho.sh"
;;
"stop")
su - pentaho -c "/home/pentaho/pentaho/server/biserver-ee/stop-pentaho.sh"
;;
*)
echo "Usage: $0 { start | stop }"
;;
esac
exit 0
```

3. Save the file, then open `/home/pentaho/pentaho/server/biserver-ee/start-pentaho.sh`.
4. Change the last `if` statement to match the this example.

```
if [ "$?" = 0 ]; then
  cd "$DIR/tomcat/bin"
  export CATALINA_OPTS="-Xms4096m -Xmx6144m -XX:MaxPermSize=256m -Dsun.rmi.
dgc.client.gcInterval=3600000 -Dsun.rmi.dgc.server.gcInterval=3600000"
  env JAVA_HOME=$_PENTAHO_JAVA_HOME sh ./startup.sh
fi
```

5. Save the file and close the text editor.
6. Make the `init` script executable.

```
chmod +x /etc/init.d/pentaho
```

7. Add the Pentaho `init` script to the standard run levels by using the `update-rc.d` command, so that it runs when the system starts, and stops when the system is shut down or rebooted. This command may not exist on your computer if it is not Debian-based. If that is the case, consult your distribution documentation or contact your distribution's support department to determine how to add `init` scripts to the default run levels.

```
update-rc.d pentaho defaults
```

Access the User Console

The User Console is a convenient web-based tool where you can perform configuration and maintenance tasks, as well as create data analysis, interactive reports, and dashboards. There are tasks you should perform when you first login to the console.

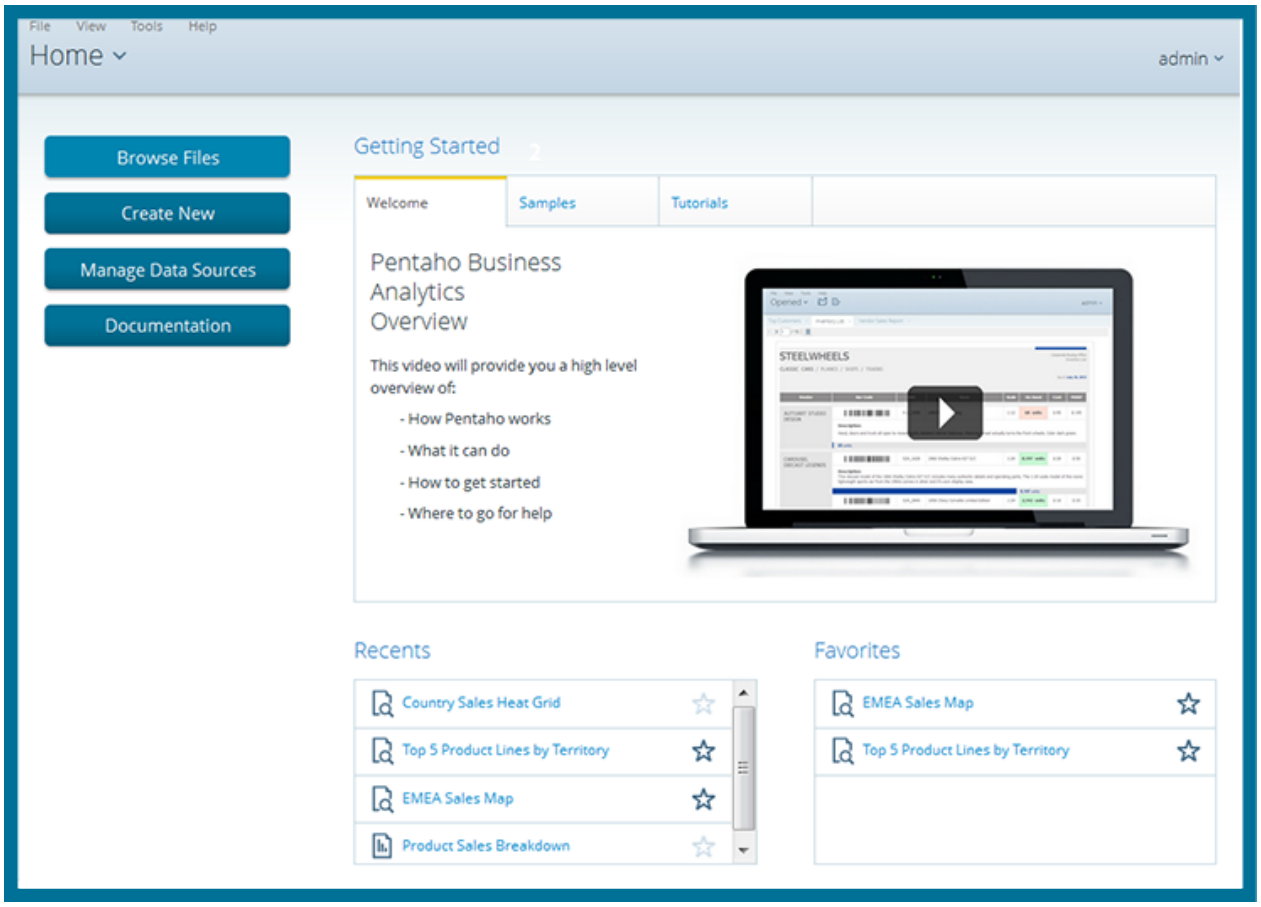
- Login
- Tour the Administration Page
- Change Administrator Password

When you are done, please go on to the next stop on the *Guide Post* graphic.

Login

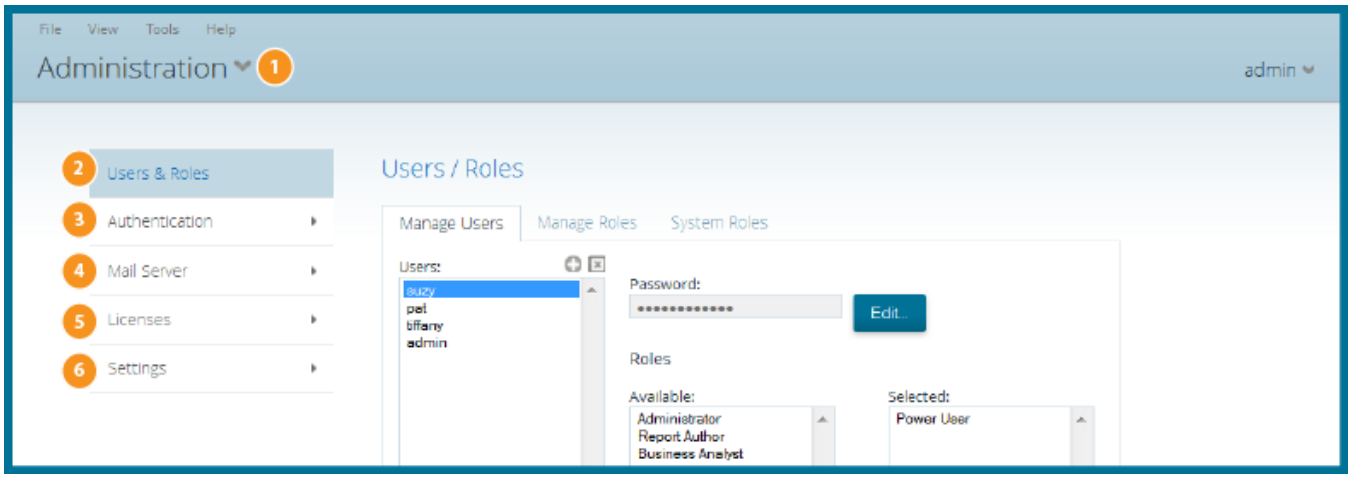
To access the User Console for the first time, login using the Pentaho administrator user name and password.

1. Start the BA Server.
2. In a browser, go to the IP address of the server and the port. For example, because the default server is localhost and the default port is 8080, the default URL is **http://localhost:8080/**. The Login dialog box appears.
3. Enter the default administrator name, `admin`.
4. Enter the default password, `password`.
5. Click **OK**. The User Console **Home** page appears.



Tour the Administration Page

The User Console has one unified place, called the **Administration** page, where people logged in with a role that has permissions to administer security can perform system configuration and maintenance tasks. If you see **Administration** in the left drop-down menu on the **User Console Home** page, you can click it to reveal menu items having to do with administration of the BA Server. If you do not have administration privileges, **Administration** does not appear on the home page.



Item	Control Name	Function
1	**Administration**	Open the Administration perspective of the User Console. The Administration perspective enables you to set up users, configure the mail server, change authentication settings on the BA Server, and install software licenses for Pentaho.
2	**Users & Roles**	Manage the Penatho users or roles for the BA Server.
3	**Authentication**	Set the security provider for the BA Server to either the default Pentaho Security or LDAP/Active Directory.
4	**Mail Server**	Set up the outgoing email server and the account used to send reports through email.
5	**Licenses**	Manage Pentaho software licenses.
6	**Settings**	Manage settings for deleting older generated files, either manually or by creating a schedule for deletion.

Change the Administrator Password

After you have logged into the User Console for the first time, it is a best practice to change the default administrator password.

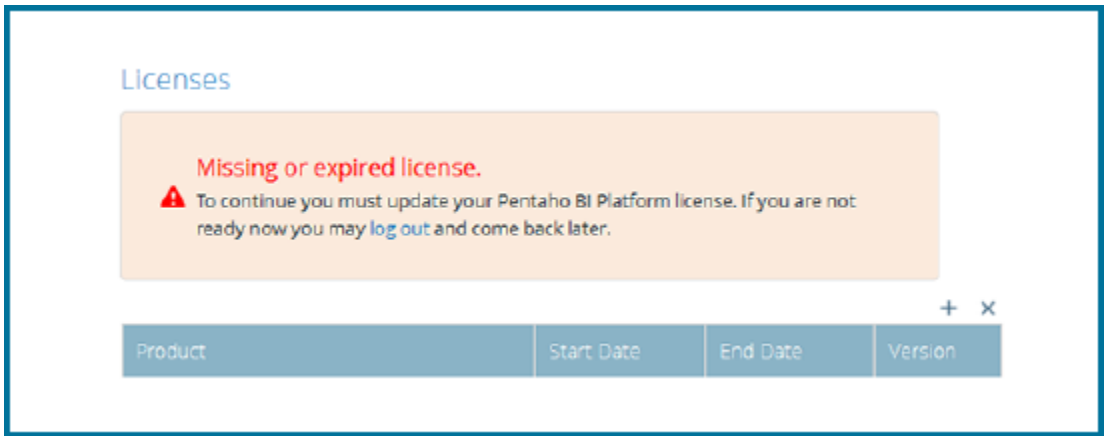
Manage License Keys for the BA Server

Your Pentaho Welcome Kit contains license keys stored in `.lic` files. Each Pentaho product has its own `.lic` file. If you did not receive your welcome kit, or if you have lost the `.lic` files, contact your Pentaho support representative. If you do not yet have a support representative, contact your Pentaho sales representative.

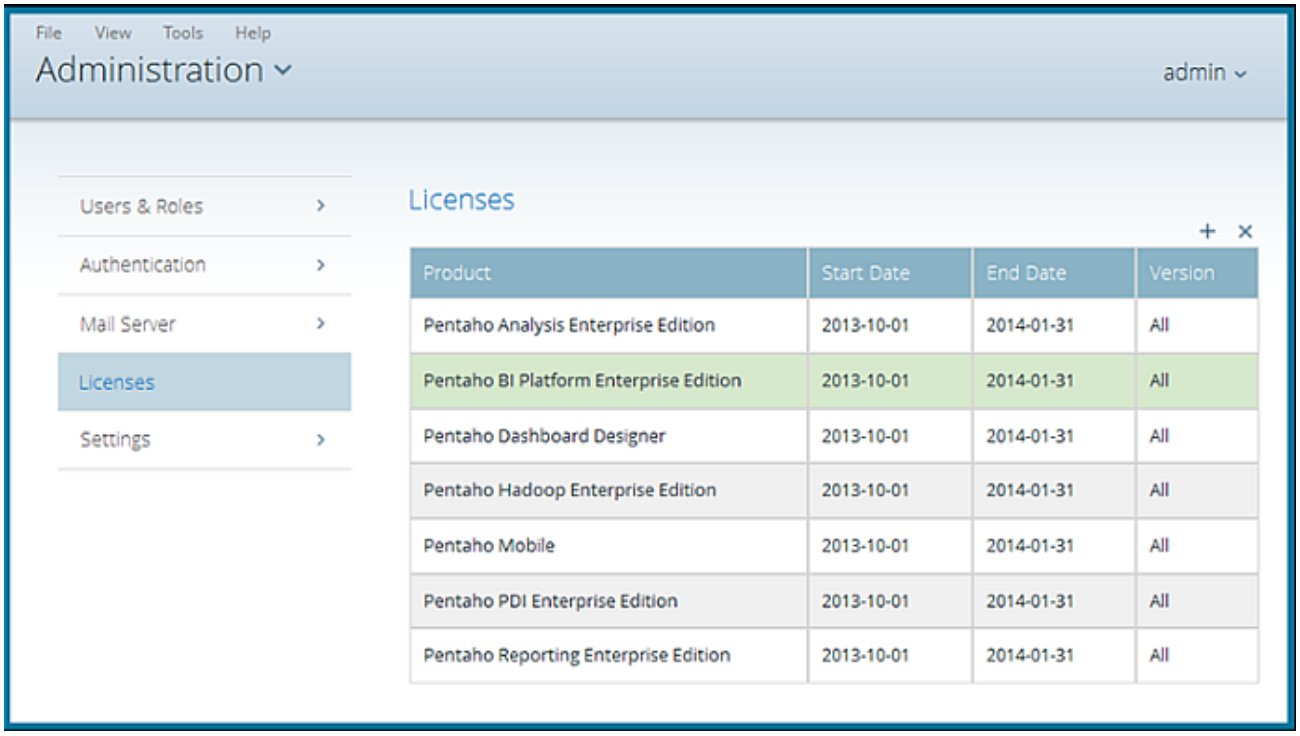
When you are done installing your licenses, please go on to the next stop on the *Guide Post* graphic.

Install License Keys

1. Copy your license files to a convenient location.
2. Log into the User Console. The first time you log in, the console takes you straight to the license page to install Pentaho licenses.



3. Click the **Add** button (+), then browse to the license files that you want to install. Double-click the license to install it. The full User Console appears after the Pentaho BI Platform license is installed.
4. Repeat the previous step until all needed licenses are installed.



Update and Delete Old License Keys

When license keys change, expire, or you purchase new Pentaho products, you may have to update the license keys.

1. Copy your new license files to a convenient location.
2. Log into the User Console.
3. Click on **Administration** in the drop-down at the top-left of the page, then click on **Licenses**. The Licenses panel appears.
4. Click to highlight the expired license, then click the **Delete** button (**x**) to delete the license. When prompted, click **Yes** to delete
5. Click the **Add** button (**+**), then browse to the license file that you want to update. Double-click the license to install it.
6. Repeat the previous steps until all needed licenses are updated.

Specify Data Connections for the BA Server

The User Console is used to define the connection to where you store data. We support accessing data stored in these ways.

- Comma separated values (CSV) files or any file that can be converted to CSV, such as spreadsheets, XML, or other semi-structured or tabular data file
- JDBC drivers , to help with database connections

- **Define Native (JDBC) Data Connections for the BA Server**
- **Define OCI Connections for the BA Server**
- **Define JNDI Connections for the BA Server**

CSV Files

To work with `.csv` files, the only required information is the location of the files.

Use `.csv` files if you want to get started quickly, have used the **Pentaho Trial Download**, are evaluating or testing, or do not have the information or expertise to configure a database connection. There is nothing you need to do now to configure these connections. Please skip this topic and go to the next stop on the *Guide Post* graphic.

JDBC Database Connections

To connect to databases, install the driver for your database, as well as define the access protocol and settings now. You can choose from these access protocols.

- Native (**JDBC**): This is a commonly used access protocol. Please see details in the Database Access Protocol Decision Table to ensure you make an informed choice.
- **JNDI**: This also is a commonly used access type. Please see details in the Database Access Protocol Decision Table to ensure you make an informed choice.
- **ODBC**: We do not support ODBC, but we make it available in case you need it. If you must use ODBC, contact Pentaho support to ensure you are successful.
- **OCI**: If you are connecting to an Oracle database, click this link to install the appropriate OCI driver and add the OCI connection now.

Table 1. Database Access Protocol Decision Table

Explore Considerations	Choose Options	
	Native (JDBC)	JNDI
Summary	Native (JDBC) connections are the easiest way to get going quickly. You specify the connection information in the User Console. The connections are controlled by the BA Server. If the connection information changes, you change it in the User Console for each connection you have defined.	JNDI connections are maintained in the application server, offering more advanced configuration options. One typical use case is you may want to hide security credentials from administrators of the Pentaho system. You specify the connection information by editing the `context.xml` file and selecting JNDI as the access type in the User Console. If the connection information changes, you change the `context.xml` file.
Expertise	Knowledge of the JDBC driver and options for your RDBMS	Knowledge of Tomcat or JBoss JNDI connection procedures and options
Time	Approximately 10 minutes	Approximately 30 minutes
Recommendation	**Use for the Pentaho Trial Download and evaluation.**	**Use for production, when the work environment is distributed in a network, or if you want to hide security credentials from Pentaho administrators.**

Define Native (JDBC) Data Connections for the BA Server

Once you have chosen to use the Native (JDBC) access protocol, there are configuration and maintenance tasks that you need to perform.

Add Drivers

The BA Server needs the appropriate driver to connect to the database that stores your data. Your database administrator, Chief Intelligence Officer, or IT manager should be able to provide the appropriate driver. If not, you can download drivers from your database vendor's website. The Supported Technologies section contains a list of drivers.

Once you have the correct driver, copy it to this directory for the BA Server: `/pentaho/server/biserver-ee/tomcat/lib/`.

There should be only one driver for your database in this directory. Ensure that there are no other versions of the same vendor's driver in this directory. If there are, back up the old driver files and remove them to avoid version conflicts. This is a concern when you are adding a driver for the same database type as your Pentaho BA repository. If you have any concerns about how to proceed, contact Pentaho support.

Driver for Microsoft SQL Server

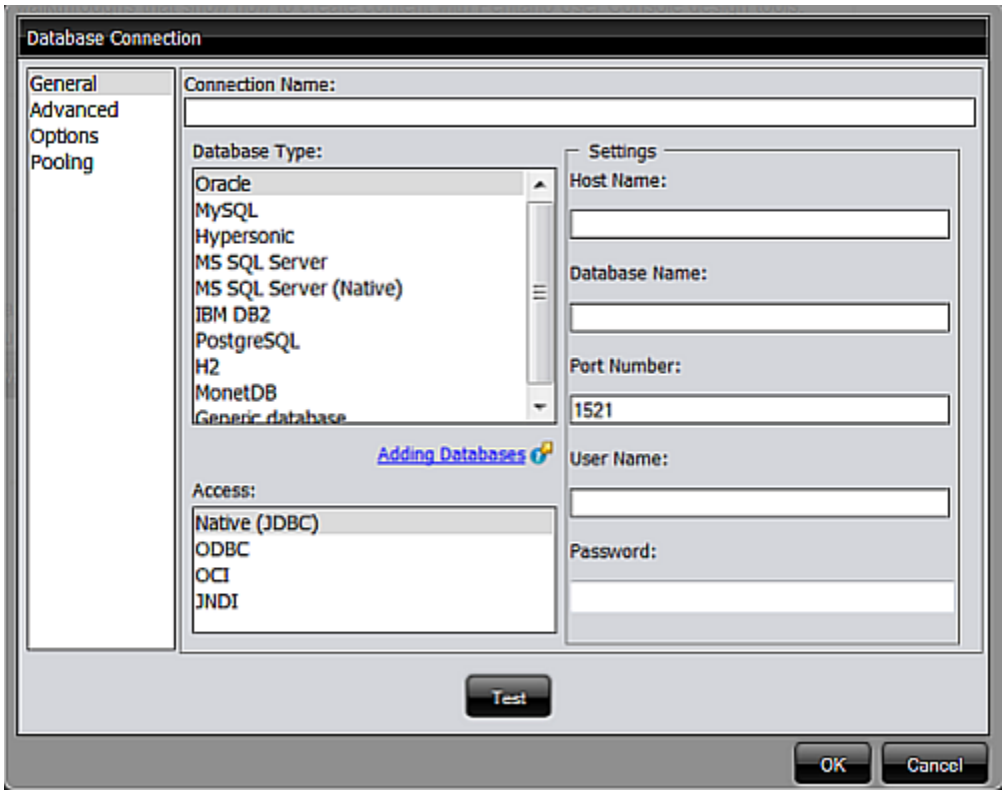
If you are using a Microsoft SQL Server (MSSQL), you might need to use an alternative, non-vendor-supported driver called JTDS. Contact Pentaho support to ensure that you are adding the correct driver.

For Microsoft Windows, most JDBC drivers support Type 2 integrated authentication through the `integratedSecurity` connection string property. To use integrated authentication, copy the `sqljdbc_auth.dll` file to all machines and directories to which you copied the JDBC driver. You can find this file in this location: `<installation directory>\sqljdbc_<version>\<language>\auth\`

If running:	Use the sqljdbc_auth.dll file here:
32-bit Java Virtual Machine (JVM) even if the operating system is version x64	x86 folder
64-bit JVM on a x64 processor	x64 folder
64-bit JVM on an Itanium processor	IA64 folder

Specify Native (JDBC) Connection Information

1. Start the web application and BA Servers, log into the User Console, then click on **File** > **Manage** > **Data Source**. The **Data Sources** dialog box appears.
2. Click the plus icon (+) on the right and select **JDBC**. The **Database Connection** dialog box appears with **General** highlighted in the left navigation pane.



3. In the **Connection Name** field, enter a name that uniquely describes this connection. The name can have spaces, but it cannot have special characters, such as #, $, %, and alike.
4. In the **Database Type** list, select the database you want to, for example, MySQL, Oracle, and so on.
5. In the **Access** list, select **Native (JDBC)**. The access protocol that appear depend on the database type you selected.
6. Under **Settings**

g. A. In the **Host Name** field, enter the name of the server that hosts the database to which you are connecting. Alternatively, you can specify the host by IP address.

h. B. In the **Database Name** field, enter the name of the database to which you are connecting.

i. C. In the **Port Number** field, enter the TCP/IP port number if it is different from the default.

j. D. Enter the **User Name** and **Password** required to access the database.

7. Click **Test**. A success message appears if the connection is established.
8. To save the connection, click **OK** twice. This connection name appears in the list of available data sources in the **Data Sources** dialog box.

Edit Existing Connections

Once a connection has been established, you can open the **Database Connection** dialog box to further refine and change aspects of the connection.

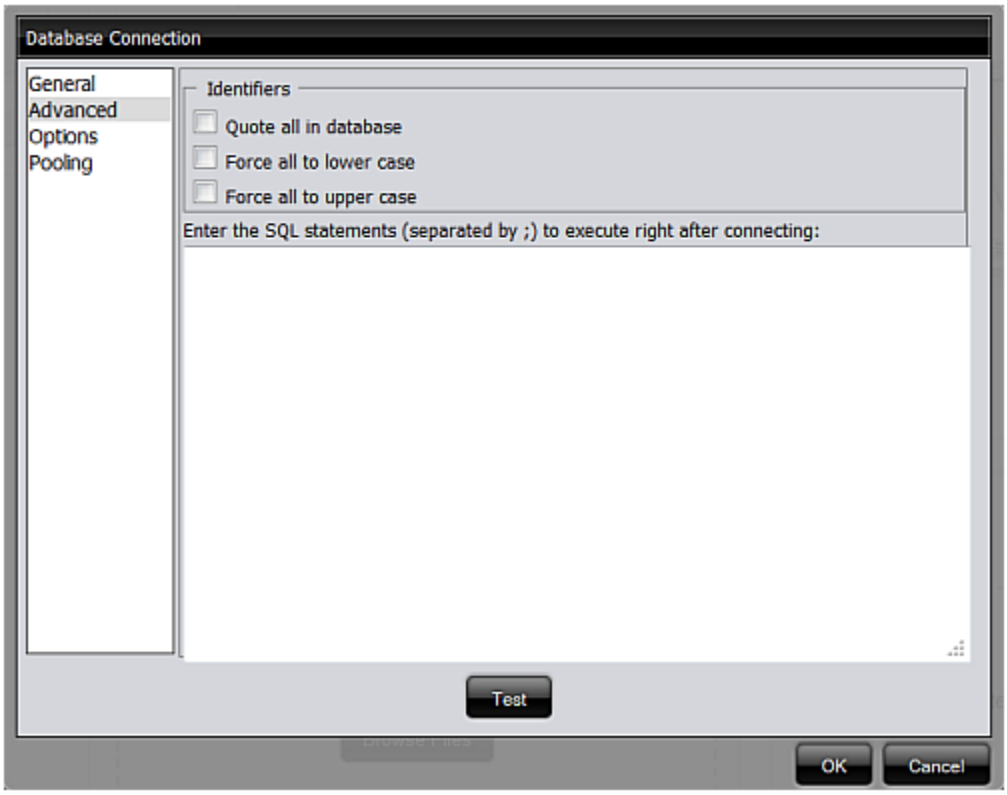
1. In the User Console, click on **File** > **Manage** > **Data Source**. The **Data Sources** dialog box appears.
2. Highlight the name of the data source you want to edit and click **Edit** (the pencil icon on the right). The **Database Connection** dialog box appears with **General** highlighted in the left navigation pane.

   To further refine your connection, or to enable and disable connection options, select items in the left navigation pane.

Specify Advanced SQL Preferences

Setting SQL preferences enables you to standardize how Pentaho software generates SQL and uses a default schema. Preferences for database table and column names are honored, regardless of how they are entered.

1. Open the Database Connection dialog box.
2. Click **Advanced** on the left. The SQL preferences appear.



3. Check the appropriate boxes and enter the SQL statements as described in this table.

Feature	Description
Quote all in database	Enables case-sensitive table names

Feature	Description
	For example MySQL is case-sensitive on Linux, but not case-sensitive on Microsoft Windows. If you quote the identifiers, the databases uses a case-sensitive table name.
Force all to lower-case	Enables the system to change the case of all database to lower-case
Force all to upper-case	Enables the system to change the case of all identifiers to upper-case
SQL statements ...	Enter the SQL statement used to initialize this connection.

4. Click **Test**. A success message appears if the connection is established.
5. To save the options, click **OK** twice.

Quoting

Pentaho uses a database-specific quoting system that allows you to use any name or character that complies with the supported databases' naming conventions.

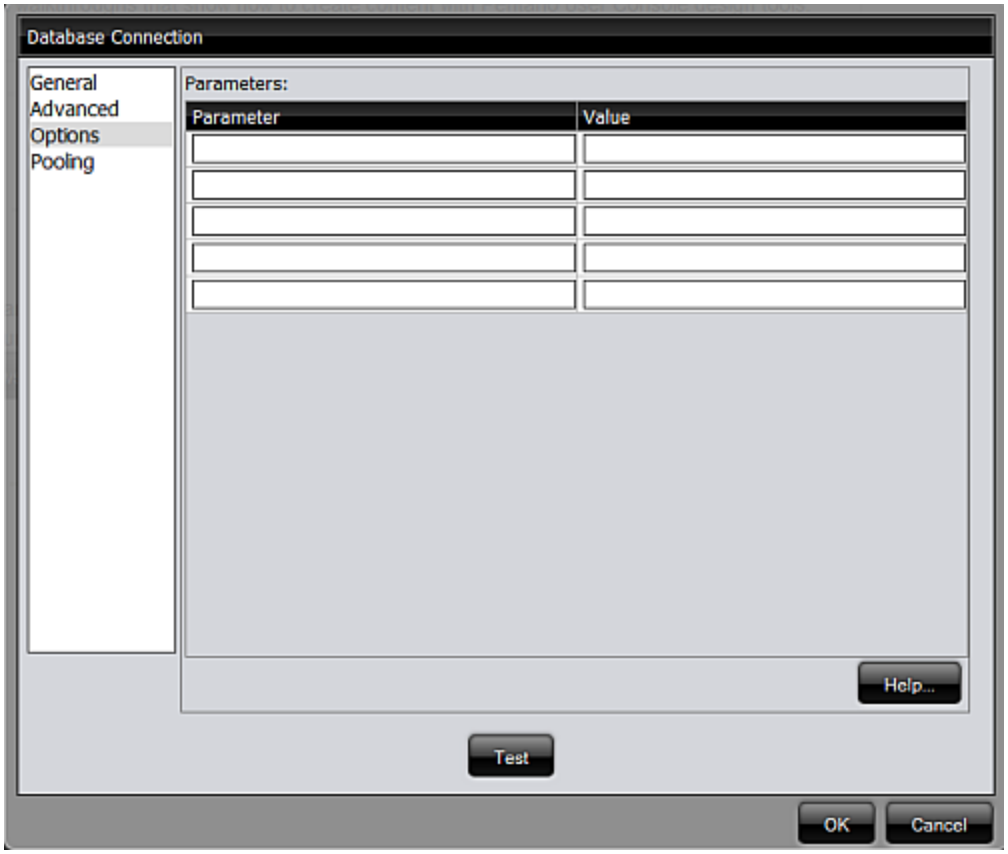
Pentaho User Console contains a list of reserved words for most of the supported databases. To ensure that quoting behaves correctly, Pentaho maintains a separation between the schema, or user/owner of a table, and the table name itself. Doing otherwise makes it impossible to correctly quote table or field names that contain one or more periods (.).

To avoid quoting-related errors, a rule stops the Pentaho software from performing quoting activity when there is a start or end quote in the table or schema name. This allows you to specify the quoting mechanism yourself.

Set Options

The driver for your specific database may have options for handling different aspects of its operation. To prepare for setting these options, refer to the driver documentation and look for the names of parameters and values you can set. Then, add them to the **Database Connections** dialog box.

1. Open the Database Connection dialog box.
2. Click **Options** on the left. The Parameter table appears.



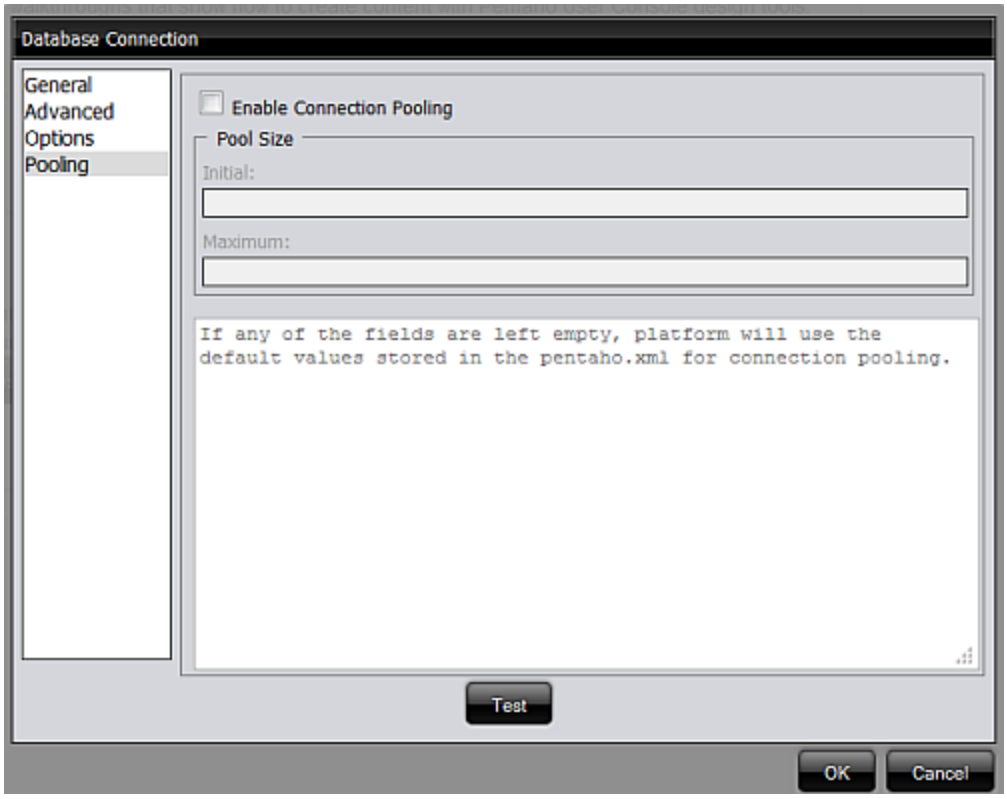
3. In the next available row in the **Parameters** table, enter a valid parameter name and its corresponding value. For JDBC database-specific configuration help, click **Help**. A new browser window opens and displays additional information about configuring the JDBC connection for the database type that is currently selected in the **General** pane.
4. Click **Test**. A success message appears if the connection is established.
5. To save the options, click **OK** twice.

Define Connection Pooling

Instead of having a connection open for each individual report or set of reports in a dashboard, you can set up a connection pool and define options that are available for your driver. For example, you might start by specifying a pool of ten or fifteen connections, and as you run reports, the unused connections drop off.

Pooling helps control database access, especially if you have dashboards that contain many reports and require a large number of connections. Pooling can also be implemented when your database licensing restricts the number of active concurrent connections.

1. Open the Database Connection dialog box.
2. Click **Pooling** on the left. Options appear for your JDBC driver. Enter the appropriate options.



This table shows an example of pooling options that might be available in a typical JDBC driver. Check your driver documentation for details.

Feature	Description
Enable Connection Pooling	Enables connection pooling.
Pool Size	Sets the **Initial** size of the connection pool; and the **Maximum** number of connections in the connection pool.
Parameters:	Click on any parameter to get a short description of that parameter. The most commonly used parameter is **validationQuery**. The parameter is slightly different depending on the RDBMS that is being connected to. • For Oracle and PostgreSQL, use "Select 1 from dual" • For MS SQL Server and MySQL, use "Select 1"

3. Click **Test**. A success message appears if the connection is established.
4. To save the options, click **OK** twice.

Delete Connections

If you no longer need a connection, you can open the **Database Connection** dialog box and delete the connection.

CAUTION:

Deleting a connection affects all reports, charts, dashboards, and other content that are associated with the connection.

1. In the User Console click on **File** > **Manage** > **Data Source**. The **Data Sources** dialog box appears.
2. Highlight the name of the data source you want to delete and click **Remove** (the x icon in the red box on the right). The data source no longer appears in the list of data sources.

Define OCI Connections for the BA Server

Once you have chosen to use the OCI access protocol, here are configuration and maintenance tasks you can perform.

- Add Drivers
- Specify Connections
- Edit Existing Connections
- Specify Advanced SQL Preferences
- Set Options
- Define Connection Pooling
- Delete Connections

Add Drivers

The BA Server needs the appropriate driver to connect to the database that stores your data. Your database administrator, Chief Intelligence Officer, or IT manager should be able to provide the appropriate driver. If not, you can download drivers from your database vendor's website. The Supported Technologies section contains a list of drivers.

Once you have the correct driver, copy it to this directory for the BA Server: `/pentaho/server/biserver-ee/tomcat/lib/`.

There should be only one driver for your database in this directory. Ensure that there are no other versions of the same vendor's driver in this directory. If there are, back up the old driver files and remove them to avoid version conflicts. This is a concern when you are adding a driver for the same database type as your Pentaho BA repository. If you have any concerns about how to proceed, contact Pentaho support.

Driver for Microsoft SQL Server

If you are using a Microsoft SQL Server (MSSQL), you might need to use an alternative, non-vendor-supported driver called JTDS. Contact Pentaho support to ensure that you are adding the correct driver.

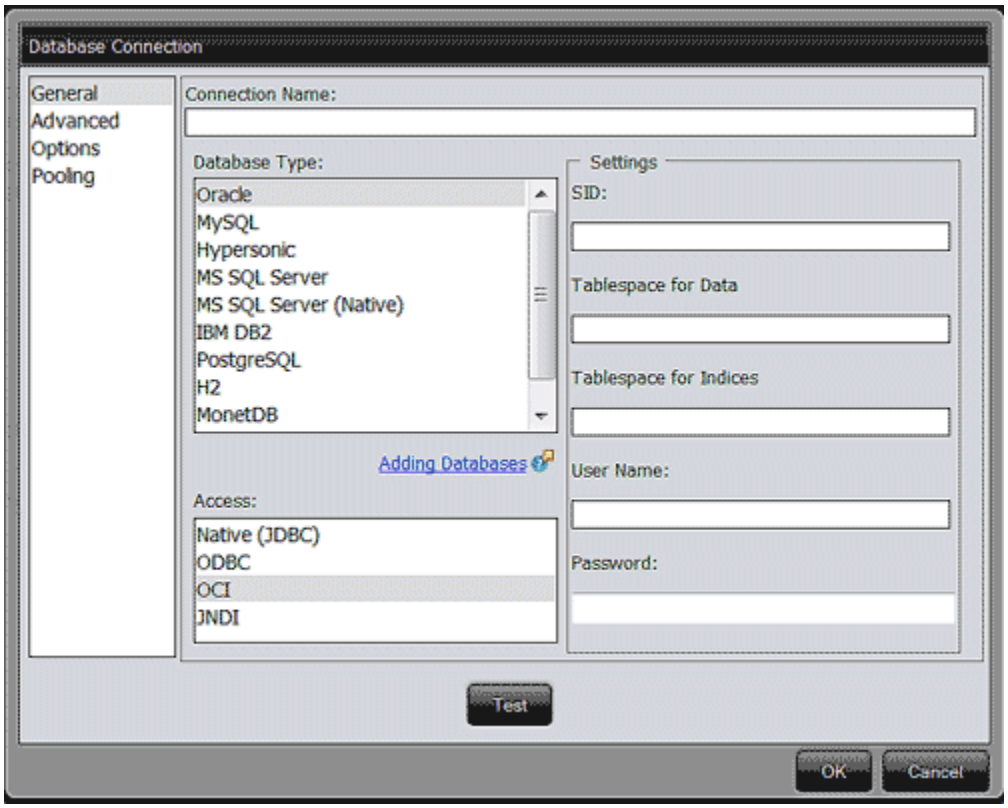
For Microsoft Windows, most JDBC drivers support Type 2 integrated authentication through the `integratedSecurity` connection string property. To use integrated authentication, copy the `sqljdbc_auth.dll` file to all machines and directories to which you copied the JDBC driver. You can find this file in this location.

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

If running:	Use the sqljdbc_auth.dll file here:
32-bit Java Virtual Machine (JVM) even if the operating system is version x64	x86 folder
64-bit JVM on a x64 processor	x64 folder
64-bit JVM on an Itanium processor	IA64 folder

Specify OCI Connections Information

1. Start the web application and BA Servers, log into the User Console, then click on **File** > **Manage** > **Data Source**. The **Data Sources** dialog box appears.
2. Click the plus icon (+) on the right and select **JDBC**. The **Database Connection** dialog box appears with **General** highlighted in the left navigation pane.



3. In the **Connection Name** field, enter a name that uniquely describes this connection. The name can have spaces, but it cannot have special characters, such as #, $, %.
4. In the **Database Type** list, select **Oracle**.
5. In the **Access** list, select **OCI**.
6. Enter **Settings** as directed by the Oracle OCI documentation.
    a. In the **SID** field, enter the Oracle system ID that uniquely identifies the database on the system.
    b. In the **Tablespace for Data** field, enter the name of the tablespace where the data is stored.
    c. In the **Tablespace for Indicies** field, enter the name of the tablespace where the indices are stored.
    d. Enter the **User Name** and **Password** required to access the database.

7. Click **Test**. A success message appears if the connection is established.
8. To save the connection, click **OK** twice. This connection name appears in the list of available data sources in the **Data Sources** dialog box. If you want to use Advanced, Options, or Pooling, refer to the Oracle OCI documentation to understand how to specify these settings.

Edit Existing Connections

Once a connection has been established, you can open the **Database Connection** dialog box to further refine and change aspects of the connection.

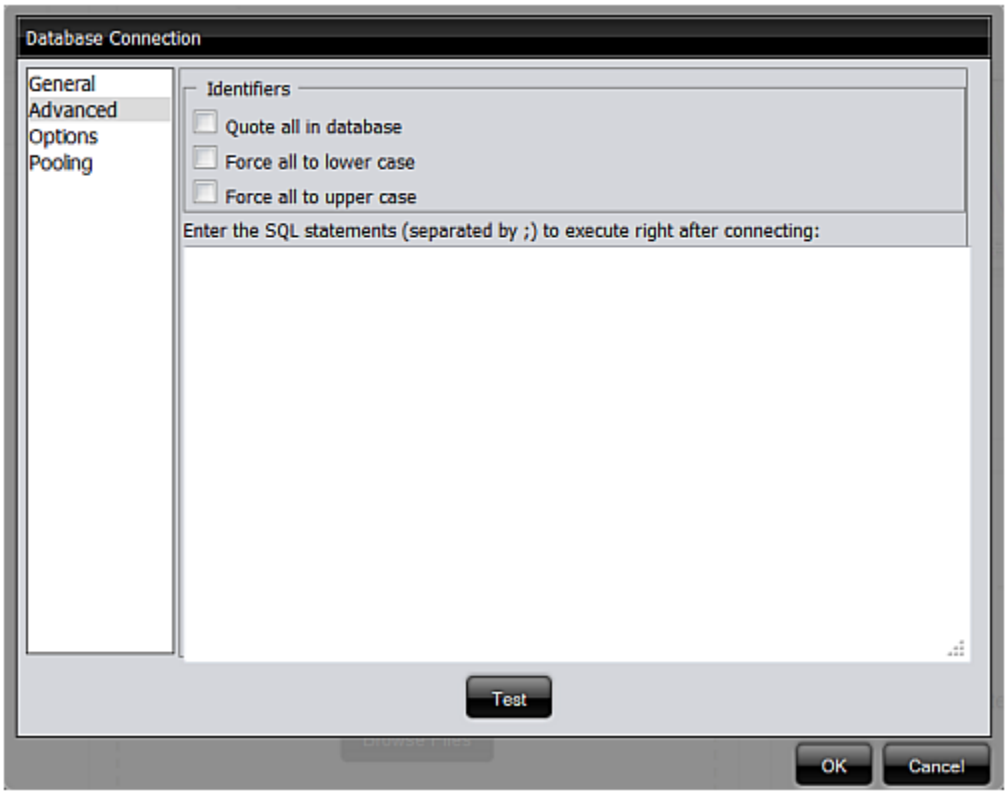
1. In the User Console, click on **File** > **Manage** > **Data Source**. The **Data Sources** dialog box appears.
2. Highlight the name of the data source you want to edit and click **Edit** (the pencil icon on the right). The **Database Connection** dialog box appears with **General** highlighted in the left navigation pane.

   To further refine your connection, or to enable and disable connection options, select items in the left navigation pane.

Specify Advanced SQL Preferences

Setting SQL preferences enables you to standardize how Pentaho software generates SQL and uses a default schema. Preferences for database table and column names are honored, regardless of how they are entered.

1. Open the Database Connection dialog box.
2. Click **Advanced** on the left. The SQL preferences appear.



3. Check the appropriate boxes and enter the SQL statements as described in this table.

Feature	Description
Quote all in database	Enables case-sensitive table names For example MySQL is case-sensitive on Linux, but not case-sensitive on Microsoft Windows. If you quote the identifiers, the databases uses a case-sensitive table name.
Force all to lower-case	Enables the system to change the case of all database to lower-case
Force all to upper-case	Enables the system to change the case of all identifiers to upper-case
SQL statements ...	Enter the SQL statement used to initialize this connection.

4. Click **Test**. A success message appears if the connection is established.
5. To save the options, click **OK** twice.

Quoting

Pentaho uses a database-specific quoting system that allows you to use any name or character that complies with the supported databases' naming conventions.

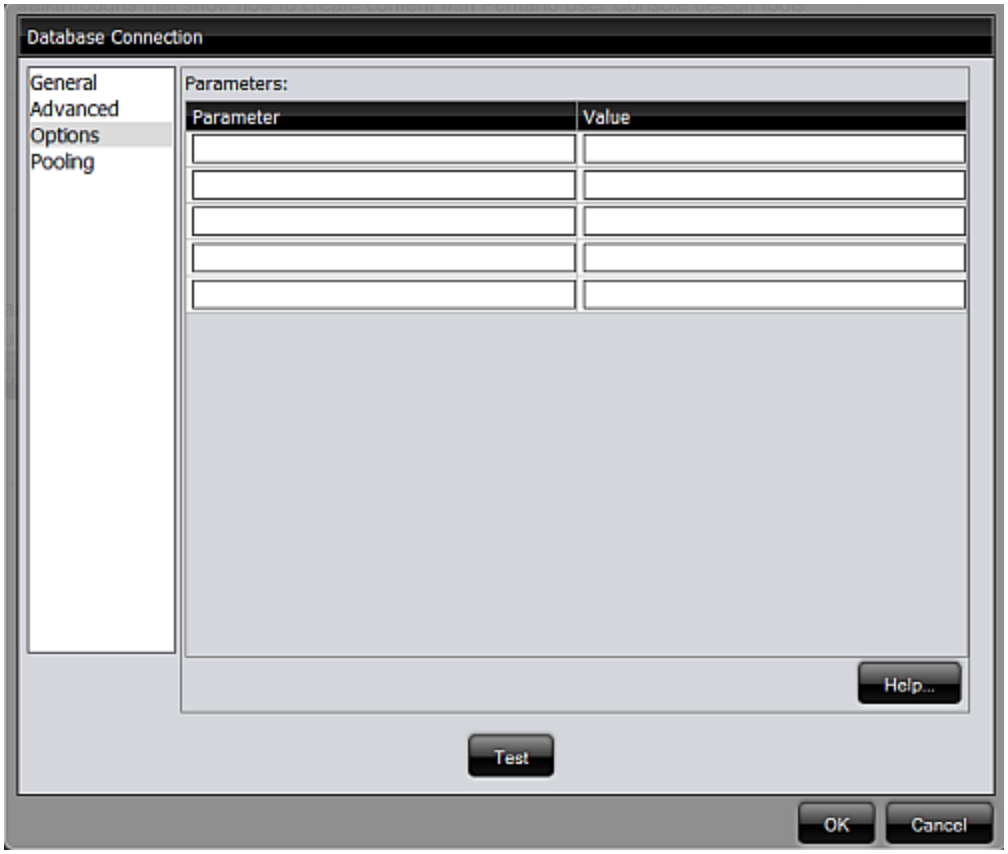
Pentaho User Console contains a list of reserved words for most of the supported databases. To ensure that quoting behaves correctly, Pentaho maintains a separation between the schema, or user/owner of a table, and the table name itself. Doing otherwise makes it impossible to correctly quote table or field names that contain one or more periods (.).

To avoid quoting-related errors, a rule stops the Pentaho software from performing quoting activity when there is a start or end quote in the table or schema name. This allows you to specify the quoting mechanism yourself.

Set Options

The driver for your specific database may have options for handling different aspects of its operation. To prepare for setting these options, refer to the driver documentation and look for the names of parameters and values you can set. Then, add them to the **Database Connections** dialog box.

1. Open the Database Connection dialog box.
2. Click **Options** on the left. The Parameter table appears.



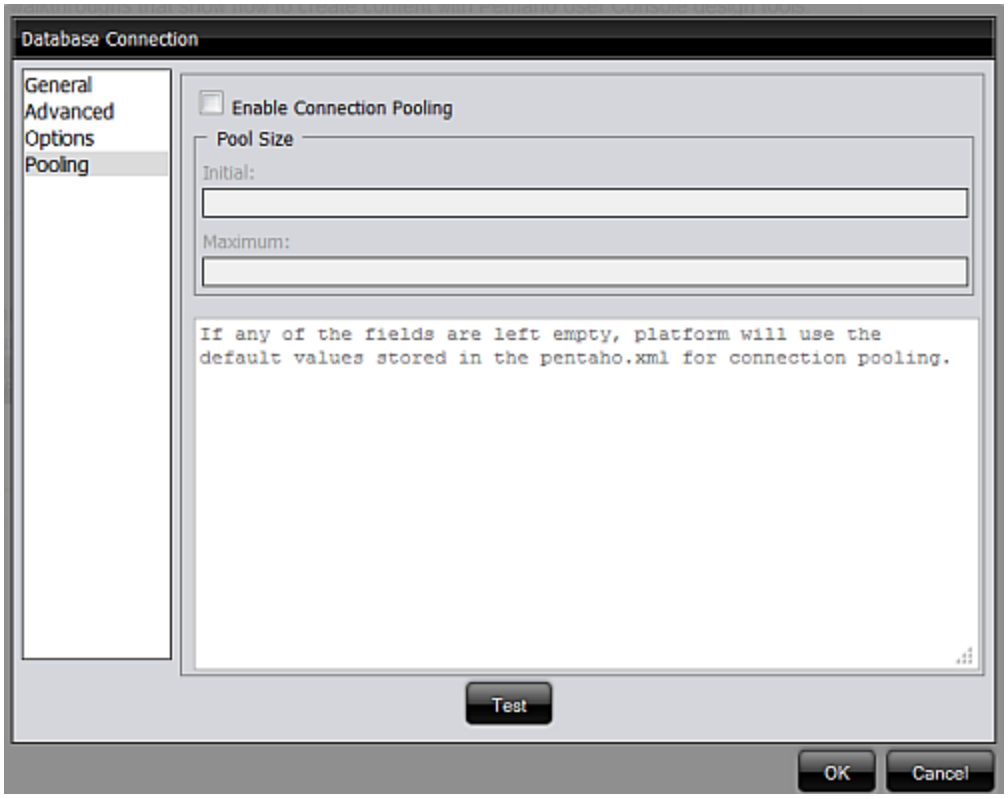
3. In the next available row in the **Parameters** table, enter a valid parameter name and its corresponding value. For JDBC database-specific configuration help, click **Help**. A new browser window opens and displays additional information about configuring the JDBC connection for the database type that is currently selected in the **General** pane.
4. Click **Test**. A success message appears if the connection is established.
5. To save the options, click **OK** twice.

Define Connection Pooling

Instead of having a connection open for each individual report or set of reports in a dashboard, you can set up a connection pool and define options that are available for your driver. For example, you might start by specifying a pool of ten or fifteen connections, and as you run reports, the unused connections drop off.

Pooling helps control database access, especially if you have dashboards that contain many reports and require a large number of connections. Pooling can also be implemented when your database licensing restricts the number of active concurrent connections.

1. Open the Database Connection dialog box.
2. Click **Pooling** on the left. Options appear for your JDBC driver. Enter the appropriate options.



This table shows an example of pooling options that might be available in a typical JDBC driver. Check your driver documentation for details.

Feature	Description
Enable Connection Pooling	Enables connection pooling.
Pool Size	Sets the **Initial** size of the connection pool; and the **Maximum** number of connections in the connection pool.
Parameters:	Click on any parameter to get a short description of that parameter. The most commonly used parameter is **validationQuery**. The parameter is slightly different depending on the RDBMS that is being connected to. • For Oracle and PostgreSQL, use "Select 1 from dual" • For MS SQL Server and MySQL, use "Select 1"

5. Click **Test**. A success message appears if the connection is established.
6. To save the options, click **OK** twice.

Delete Connections

If you no longer need a connection, you can open the **Database Connection** dialog box and delete the connection.

CAUTION:
Deleting a connection affects all reports, charts, dashboards, and other content that are associated with the connection.

1. In the User Console click on **File** > **Manage** > **Data Source**. The **Data Sources** dialog box appears.
2. Highlight the name of the data source you want to delete and click **Remove** (the x icon in the red box on the right). The data source no longer appears in the list of data sources.

Define JNDI Connections for the BA Server

Once you have chosen to use the JNDI access protocol, here are configuration tasks you can perform.

- Add Drivers
- Specify Connections
- Tomcat JNDI Connections
- JBoss JNDI Connections

Add Drivers

The BA Server needs the appropriate driver to connect to the database that stores your data. Your database administrator, Chief Intelligence Officer, or IT manager should be able to provide the appropriate driver. If not, you can download drivers from your database vendor's website. The Supported Technologies section contains a list of drivers.

Once you have the correct driver, copy it to this directory for the BA Server: `/pentaho/server/biserver-ee/tomcat/lib/`.

There should be only one driver for your database in this directory. Ensure that there are no other versions of the same vendor's driver in this directory. If there are, back up the old driver files and remove them to avoid version conflicts. This is a concern when you are adding a driver for the same database type as your Pentaho BA repository. If you have any concerns about how to proceed, contact Pentaho support.

Driver for Microsoft SQL Server

If you are using a Microsoft SQL Server (MSSQL), you might need to use an alternative, non-vendor-supported driver called JTDS. Contact Pentaho support to ensure that you are adding the correct driver.

For Microsoft Windows, most JDBC drivers support Type 2 integrated authentication through the `integratedSecurity` connection string property. To use integrated authentication, copy the `sqljdbc_auth.dll` file to all machines and directories to which you copied the JDBC driver. You can find this file in this location.

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

If running:	Use the sqljdbc_auth.dll file here:
32-bit Java Virtual Machine (JVM) even if the operating system is version x64	x86 folder
64-bit JVM on a x64 processor	x64 folder

If running:	Use the sqljdbc_auth.dll file here:
64-bit JVM on an Itanium processor	IA64 folder

Specify JNDI Connection Information

Instructions for adding JNDI vary depending on which web application server you are using.

- If you used the **Pentaho Trial Download**, Installation Wizard, or Archive Installation, you are using a Tomcat web application server.
- If you used the Manual Installation, you could be using either Tomcat or JBoss.

Tomcat JNDI Connections

1. Stop the Tomcat and BA servers.
2. Consult your database documentation to determine the class name and connection string for your database.
3. Edit the `/tomcat/webapps/pentaho/WEB-INF/web.xml` file.
4. At the end of the `<web-app>` element, in the same part of the file where you see `<!-- insert additional resource-refs -->`, add this XML snippet.

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

   Change the `description` and `res-ref-name` nodes, as well as any others that apply to your situation and fit your database. You may need to consult http://tomcat.apache.org/tomcat-6.0-doc/jndi-datasource-examples-howto.html to see if there are other things to consider.
5. Save and close the `web.xml` file.
6. Edit the `/tomcat/conf/context.xml` with a text editor. Alternatively, you can modify the `/tomcat/webapps/pentaho/META-INF/context.xml` file if you want this data connection to be available only to the BA Server. Adding JNDI connections to the `context.xml` makes them available to all of the webapps deployed to this Tomcat instance.
7. Anywhere inside of the `<Context>` element, add this XML snippet. Then save and close the `context.xml` file.

   ```
   <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"
/>
```

This example shows a simple PostgreSQL configuration. Replace the `Resource name`, `username`, `password`, `driverClassName`, and `url` parameters, or any relevant connection settings, to match your database connection information and the details you supplied in the `web.xml` file earlier.

8. Delete the `pentaho.xml` filed located in the `/tomcat/conf/catalina/directory`. The `pentaho.xml` is a cached copy of the `context.xml` file you modified. Since the cache is not usually configured to update frequently, you have to delete the `pentaho.xml` file and let Tomcat recreate it when it starts up.
9. Start the Tomcat and BA Server.

Tomcat can now properly connect to your data. Please go on to the next stop on the *Guide Post* graphic.

Install JDBC Driver as a Module in JBoss

In JBoss, JDBC driver information is stored in a module, which is an XML file that you create. You must download the JDBC driver software component to the correct directory, then create `module.xml` files for each database. You need to create a file for the database that hosts the BA Repository (either PostgreSQL, MySQL, or Oracle).

1. Navigate to the `pentaho/server/biserver-ee/<your jboss installation directory>/modules/system/layers/base/org` folder and create one of the following paths for the database on which you are hosting the BA Repository.
   - **PostgreSQL:** `postgresql/main`
   - **MySQL:** `mysql/main`
   - **Oracle:** `oracle/main`
5. Download the supported JDBC driver for your DI Repository database to the `postgresql/main`, `mysql/main`, or `oracle/main` directories (which ever one you created). See the JDBC Drivers Reference for a list of supported drivers.
6. In the `postgresql/main`, `mysql/main`, or `oracle/main` (which ever one you created), do the following things.
   a. Use an editor to create a text file named `module.xml`.
   b. Copy the following code to the `module.xml` file, then modify it so that the name of the JDBC driver you just downloaded appears.
      - **PostgreSQL:** If you are using PostgreSQL, copy the following code in the `module.xml` file. Replace the name of the `resource-root path` parameter with the name of the JDBC driver you downloaded.

```
    <?xml version="1.0" encoding="UTF-8"?>
    <module xmlns="urn:jboss:module:1.0" name="org.postgresql">
        <resources>
            <resource-root path="[Name of JDBC Jar You Downloaded
Here]"/>
```

```
        </resources>
        <dependencies><module name="javax.api"/></dependencies>
    </module>
```

- **MySQL:** If you are using MySQL, copy the following code in the `module.xml` file. Replace the name of the `resource-root path` parameter with the name of the JDBC driver you downloaded.

```
    <?xml version="1.0" encoding="UTF-8"?>
    <module xmlns="urn:jboss:module:1.0" name="org.mysql">
        <resources>
            <resource-root path="[Name of JDBC Jar You Downloaded
Here]"/>
        </resources>
        <dependencies><module name="javax.api"/></dependencies>
    </module>
```

- **Oracle:** If you are using Oracle, copy the following code in the `module.xml` file. Replace the name of the `resource-root path` parameter with the name of the JDBC driver you downloaded.

```
    <?xml version="1.0" encoding="UTF-8"?>
    <module xmlns="urn:jboss:module:1.0" name="org.oracle">
        <resources>
            <resource-root path="[Name of JDBC Jar You Downloaded
Here]"/>
        </resources>
        <dependencies><module name="javax.api"/></dependencies>
    </module>
```

f. Save and close the `module.xml` file.

JBoss JNDI Connections

1. Stop the JBoss and BA servers.
2. Consult your database documentation to determine the class name and connection string for your database.
3. Use a text editor to open the `pentaho/server/biserver-ee/jboss/standalone/configuration/standalone.xml` file.
4. Append the file to include a definition for each of your data sources. Note that you should consult your JBoss documentation for proper setup information.

```
                <datasource jndi-name="java:jboss/datasources/Hibernate"
pool-name="hibpool" enabled="true" jta="true" use-java-context="true" use-
ccm="true">
                        <connection-url>
```

```
                        jdbc:postgresql://localhost:5432/hibernate
                    </connection-url>
                    <driver-class>
                        org.postgresql.Driver
                    </driver-class>
                    <driver>
                        org.postgresql
                    </driver>
                    <pool>
                        <prefill>
                            false
                        </prefill>
                        <use-strict-min>
                            false
                        </use-strict-min>
                        <flush-strategy>
                            FailingConnectionOnly
                        </flush-strategy>
                    </pool>
                    <security>
                        <user-name>
                            hibuser
                        </user-name>
                        <password>
                            password
                        </password>
                    </security>
                </datasource>
                <datasource jndi-name="java:jboss/datasources/Quartz"
pool-name="quartzpool" enabled="true" jta="true" use-java-context="true"
use-ccm="true">
                    <connection-url>
                        jdbc:postgresql://localhost:5432/quartz
                    </connection-url>
                    <driver-class>
                        org.postgresql.Driver
                    </driver-class>
                    <driver>
                        org.postgresql
                    </driver>
```

```
                        <pool>
                            <prefill>
                                false
                            </prefill>
                            <use-strict-min>
                                false
                            </use-strict-min>
                            <flush-strategy>
                                FailingConnectionOnly
                            </flush-strategy>
                        </pool>
                        <security>
                            <user-name>
                                pentaho_user
                            </user-name>
                            <password>
                                password
                            </password>
                        </security>
                    </datasource>
                    <datasource jndi-name="java:jboss/datasources/Audit"
pool-name="auditpool" enabled="true" jta="true" use-java-context="true" use-
ccm="true">
                        <connection-url>
                            jdbc:postgresql://localhost:5432/hibernate
                        </connection-url>
                        <driver-class>
                            org.postgresql.Driver
                        </driver-class>
                        <driver>
                            org.postgresql
                        </driver>
                        <pool>
                            <prefill>
                                false
                            </prefill>
                            <use-strict-min>
                                false
                            </use-strict-min>
                            <flush-strategy>
```

```
                                FailingConnectionOnly
                            </flush-strategy>
                        </pool>
                        <security>
                            <user-name>
                                pentaho_user
                            </user-name>
                            <password>
                                password
                            </password>
                        </security>
                    </datasource>
 <datasource jndi-name="java:jboss/datasources/Operations_Mart" pool-
name="Operations_Mart" enabled="true" jta="true" use-java-context="true"
use-ccm="true">
                        <connection-url>
                            jdbc:postgresql://localhost:5432/hibernate
                        </connection-url>
                        <driver-class>
                            org.postgresql.Driver
                        </driver-class>
                        <driver>
                            org.postgresql
                        </driver>
                        <pool>
                            <prefill>
                                false
                            </prefill>
                            <use-strict-min>
                                false
                            </use-strict-min>
                            <flush-strategy>
                                FailingConnectionOnly
                            </flush-strategy>
                        </pool>
                        <security>
                            <user-name>
                                pentaho_user
                            </user-name>
                            <password>
```

```
                                password
                            </password>
                        </security>
                    </datasource>
```

5. If your environment (e.g. port numbers, IP address), solution repository, or database password and user name information differs from the code in the above example, modify it to match your specifications.
6. Add the driver definition in the driver section of the file. Here is an example of the PostgreSQL driver definition. If you are using MySQL or Oracle, modify the driver name, module, and data source class accordingly.

```
<driver name="org.postgresql" module="org.postgresql">
 <xa-datasource-class>
     org.postgresql.xa.PGXADataSource
     </xa-datasource-class>
 </driver>
```

7. Close and save the `standalone.xml`file.
8. Start the JBoss and BA Server.

Remove JNDI Resource References in JBoss

Because JBoss has its own mechanism for referencing JNDI data sources, the resource-references in the `web.xml` file located in the `pentaho.war` are not needed. You must remove these resource-references for the BA Server to operate properly.

1. Navigate to the `pentaho/server/biserver-ee/<your jboss installation directory>/standalone/deployments` directory.
2. Use a zip extraction utility (such as 7-Zip, Winzip, or Archive) to view the contents of the `pentaho.war` file. Do not unzip or extract the contents of the file.
3. Navigate to the `WEB-INF` directory and open the `web.xml` file in a text editor.
4. Delete all `<resource-ref>` tagged entries including everything between the `<resource-ref>` and `</resource ref>` tags.
5. Save and close the file.
6. The zip extraction utility that you used might show a prompt that asks whether you would like to update the file in the **pentaho.war** archive. If this happens, confirm that you would like to do this.

Update JNDI Data Source Reference to Conform to JBoss Standards

Update these files so that referenced JNDI datasources conform to JBoss standards.

1. Use a text editor to open the `pentaho/server/biserver-ee/pentaho-solutions/system/quartz/quartz.properties` file.
2. Change the `org.quartz.dataSource.myDS.jndiURL` value to `jboss/datasources/Quartz`, then save and close the file.
3. Use a text editor to open the `pentaho/server/biserver-ee/pentaho-solutions/system/audit_sql.xml` file.

4. Change the JNDI value to `jboss/datasources/Hibernate`, then save and close the file.
5. Use a text editor to open the `pentaho/server/biserver-ee/pentaho-solutions/system/data-access/settings.xml` file.
6. Change the `data-access-staging-jndi` value to `jboss/datasources/Hibernate`, then save and close the file.
7. Open the `pentaho/server/biserver-ee/pentaho-solutions/system/audit/dialects/` folder for your appropriate database. Use the text editor to open each file and make the following changes, using your appropriate database:
    - Change `<database>Audit</database>` to `<database>jboss/datasources/Audit</database>`.
    - Change `<database>Hibernate</database>` to `<database>jboss/datasources/Hibernate</database>`.

JBoss can now properly connect to your data. Please go on to the next stop on the *Guide Post* graphic.

Define Security for the BA Server

Using security is a best practice, but is not required. If you want to get started quickly or do not have information about your user community, skip this for now and go on to the next stop on the *Guide Post* graphic. You can always come back to it later.

We support two different security options: Pentaho Security or advanced security providers, such as LDAP, Single Sign-On, or Microsoft Active Directory. This table can help you choose the option that is best for you.

Table 1. Security Decision Table

Explore Considerations	Choose Options	
	Pentaho Security	Advanced Security Providers—LDAP, Single Sign-On, or Microsoft Active Directory
Summary	Pentaho Security is the easiest way to configure security quickly. The User Console enables you to define and manage users and roles. The BA Server controls which users and roles can access web resources through the User Console or resources in the Pentaho BA repository. Pentaho Security works well if you do not have a security provider or if you have a user community with less than 100 users.	If you are already using a security provider, such as LDAP, Single Sign-On, or Microsoft Active Directory, you can use the users and roles you have already defined with Pentaho. Your security provider controls which users and roles can access Pentaho web resources through the User Console or resources in the BA repository. Advanced security scales well for production and enterprise user communities.
Expertise	Knowledge of your user community and which users should have which roles in the Pentaho system. Knowledge about security in general is *not* required.	Knowledge of your user community and which users should have which roles in the Pentaho system. Knowledge about your particular security provider and its options is required.

Explore Considerations	Choose Options	
	Pentaho Security	Advanced Security Providers—LDAP, Single Sign-On, or Microsoft Active Directory
Time	It takes approximately 5 minutes per user and role to configure Pentaho Security.	It takes approximately 1 hour to configure the BA Server to use your existing security provider.
Recommendation	**Recommended for the Pentaho Trial Download, evaluating, and rapid development.**	**Recommended for production.**

- **Use Pentaho Security**

Use Pentaho Security

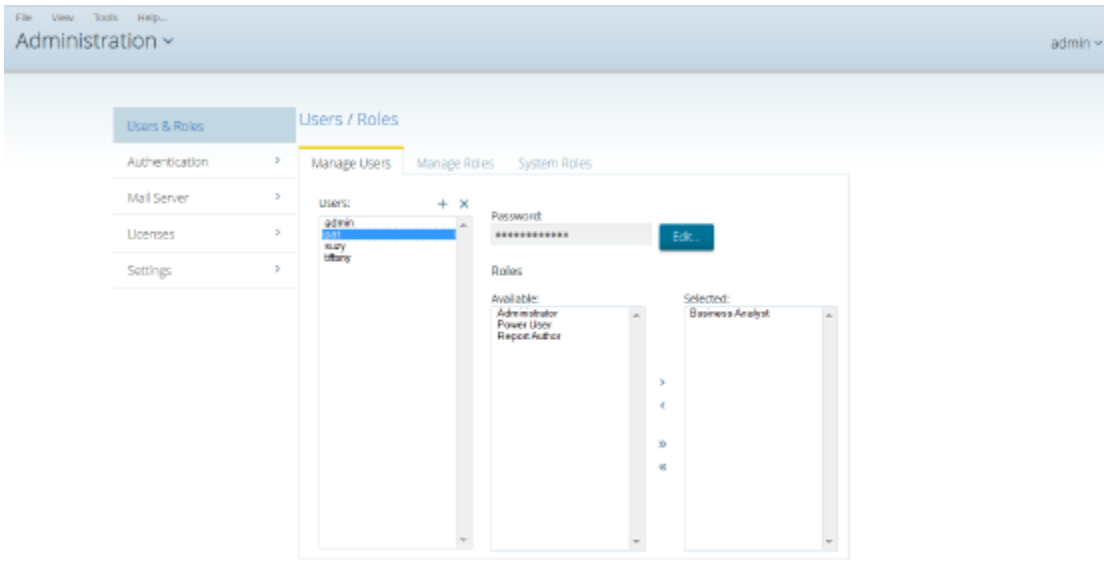
If you choose to use Pentaho Security as your security provider, you define users and roles through the User Console. The Default Users and Roles section provides an overview of the out-of-box users and roles, along with the permissions that are included with each role. Permissions can be further refined on the file- or folder-level from the **Browse** perspective of the User Console.
Before changing security settings, play it safe and back up these relevant files.

- If you installed using the Installation Wizard or the Archive Installation, back up the Pentaho Business Analytics or BA Server directories.
- If you installed manually, back up the Pentaho `.war` and solutions.

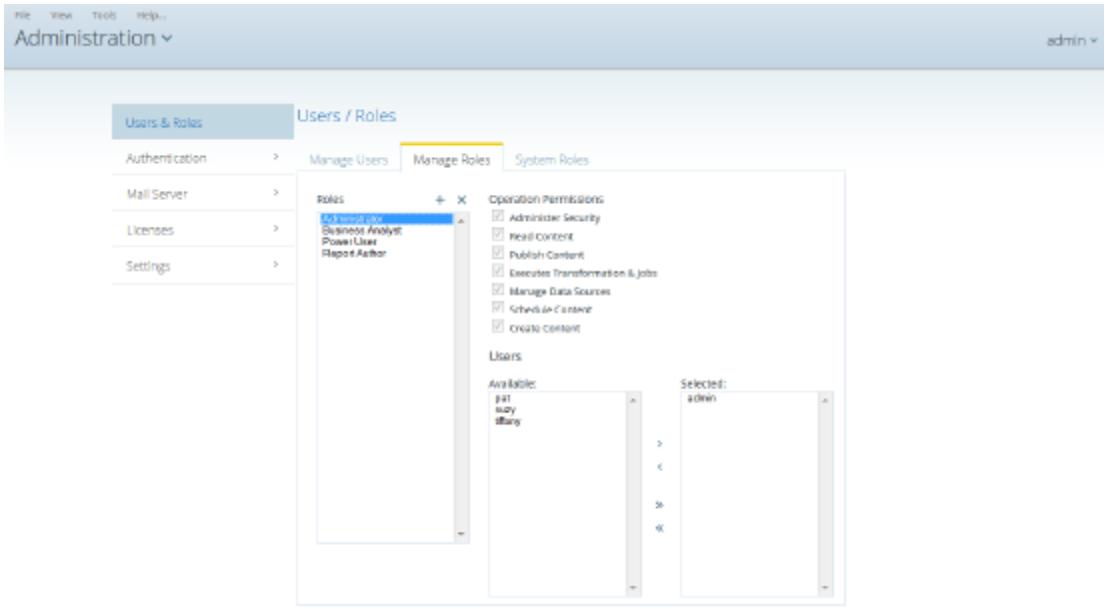
When you are done, please go on to the next stop on the *Guide Post* graphic.

Default Users and Roles

Viewing default users and roles gives you an idea of how you can define your specific users and roles. To view the default users and roles, log into the User Console, click the **Administration** perspective link on the right, then **Users and Roles** from the items on the left, and the **Manage Users** tab. Highlighting a user in the **Users** list shows which roles are available for that user, as well as which role is currently defined for that user.



The **Manage Roles** tab shows similar information as the **Manage Users** tab, with the roles listed in the pane on the left, and the associated **Operation Permissions** for each role listed on the right.



Each default role and user comes with a standard set of permissions. These roles are added for your convenience and can be removed or altered based on your needs.

The default role for all users is **Authenticated**. If you want to restrict permissions, the **Authenticated** role must be restricted or the Authenticated role must be removed from the user.

Table 1. Default Pentaho Security Settings

Out-of-Box Role	Out-of-Box User	Default Operation Permissions
Administrator	admin	• Administer Security • Read Content • Publish Content • Manage Data Sources • Executes Transformation & Jobs • Schedule Content • Create Content
Business Analyst Role	pat	• Publish Content
Power User Role	suzy	• Read Content • Publish Content • Schedule Content • Create Content
Report Author Role	tiffany	• Publish Content

Out-of-Box Role	Out-of-Box User	Default Operation Permissions
		• Schedule Content

Each operation permission gives a specific set of permissions for Pentaho tools and the BA Server.

Table 2. Operation Permissions Defined

Operation Permission	Definition
Administer Security	The default Administrator role automatically conveys all operation permissions to users assigned to that role, even if the check box next to it is cleared. This includes the Read and Create Content permissions, which are required for accessing the Administration perspective. • Gives access to the Administration perspective of the User Console. • Allows access to and the ability to manage all content in the Browse perspective. • Allows the ability to view and work with all user schedules in the Schedules perspective. • Gives the ability to create server block out times in the Schedules perspective.
Read Content	• Gives the user the ability to view content in the Browse perspective. • Gives the user the ability to view content through the File > Open dialog.
Publish Content	This permission includes tools such as Report Designer, Agile BI, Schema Workbench, and Metadata Editor. • Allows client tools to store reports or data models in the Pentaho repository.
Manage Data Sources	• Allows the user to create, edit, or delete new data sources. • Gives the user the ability to see a list of data sources that are used to create reports or dashboards. Note: This operation permission does not include Metadata data sources. This Metadata Security article gives specific information on how to give permissions to manage Metadata data sources.
Executes Transformation & Jobs	• Enables the Run, Preview, Debug, Reply, and Verify buttons and menu entries in Spoon, Kitchen, Pan, and Carte. • Allows the user to save, copy, or schedule Transformations and Jobs. • Gives the user the ability to use export buttons and associated menu entries.

Operation Permission	Definition
Schedule Content	• Allows the user to schedule reports and content. • Gives the user the ability to view, edit, or delete their own schedules using the Schedules perspective.
Create Content	• Allows the user to create, import, delete, and save reports to the repository. • Gives the user the ability to see a list of data sources that are used to create reports or dashboards.

Add Users

1. Click on the **Administration** perspective link on the upper right toolbar of the console, then click on **Users & Roles**. The **Users & Roles** interface appears.
2. Make sure the **Manage Users** tab is selected, then click the plus (**+**) sign above the list of users. The **New User** dialog box appears.
3. Type to enter a new **User Name**, **Password**, and **Confirm Password**, then click **OK**.

The new user account is active, and appears in the **Users** list.

Change User Passwords

1. Click on the **Administration** perspective link on the upper right toolbar of the console, then click on **Users & Roles**. The **Users & Roles** interface appears.
2. Make sure the Manage **Users** tab is selected. From the **Users** list, click to select the user whose password you want to edit. The user's information populates to the right of the **Users** field.
3. Click **Edit**, then enter and confirm the new password. Click **OK**.

The password is changed and the user is able to login with the new password.

Delete Users

1. Click on the **Administration** perspective link on the upper right toolbar of the console, then click on **Users & Roles**. The **Users & Roles** interface appears.
2. Make sure the **Manage Users** tab is selected, then in the **Users** field, click to select the user or users you want to delete from the server.
3. Click the **x** to delete the user or users. The **Delete User** confirmation dialog box appears.
4. Click **Yes, Delete** to delete the user(s) and refresh the user list.

The selected user accounts are deleted and the users are no longer able to log into the BA Server.

Assign Users to Roles

1. Click on the **Administration** perspective link on the upper right toolbar of the console, then click on **Users & Roles**. The **Users & Roles** interface appears.

2. Make sure the **Manage Users** tab is selected, then click to highlight the user from the **Available** user list that you want to associate with a role.
3. In the **Role Available** list, click to highlight the role that you want to associate with the selected user.
4. Click the right arrow to move the role to the **Role Selected** list.
5. You can remove a role from the **Role Selected** list by highlighting that role and clicking on the left arrow. The role moves from the **Role Selected** to **Role Available** list, and the user no longer has the permissions associated with that role.

The user now has all of the permissions associated with the role in the **Role Selected** list.

Add Roles

1. Click on the **Administration** perspective link on the upper right toolbar of the console, then click on **Users & Roles**. The **Users & Roles** interface appears.
2. Make sure the **Roles** tab is selected, then click the plus (**+**) sign above the list of roles. The **New Role** dialog box appears.
3. Type to enter a new **Name** for the role, then click **OK**.

The new role is created, and appears in the **Available** roles list.

Assign Permissions to Roles

1. After you add a new role, you need to assign operation permissions to it.
2. Make sure that the role is highlighted in the **Roles** list.
3. Assign permissions to the role by selecting from the **Operation Permissions** list to the right.

The role has permissions assigned to it, and users associated with that role have those permissions.

Delete Roles

1. Click on the **Administration** perspective link on the upper right toolbar of the console, then click on **Users & Roles**. The **Users & Roles** interface appears.
2. Make sure the **Roles** tab is selected, then from the **Available** field, click to select the role or roles you want to delete from the server.
3. Click the **x** to delete the role(s). The **Delete Role** confirmation dialog box appears.
4. Click **Yes** to delete the role(s) and refresh the role list.

The selected role is deleted and is no longer available on the server. The users that were associated with that role are no longer associated with that role. This will not affect the other roles assigned to users.

If users have only one role assigned to them and that role is deleted, then the users have no role assigned to them. The default role is **Authenticated** and all users have that role unless you remove it.

Assign Roles to Users

1. Click on the **Administration** perspective link on the upper right toolbar of the console, then click on **Users & Roles**. The **Users & Roles** interface appears.
2. Make sure the **Roles** tab is selected, then click to highlight the role from the **Available** roles list that you want to associate with a user or users.
3. In the **Members Available** list, click to highlight the user or users that you want to associate with the selected role.

4. Click the right arrow to move the selected users to the **Members Selected** list. You can click the double-right arrow to move all users from the **Members Available** to the **Members Selected** list.
5. You can remove users from the **Members Selected** list by highlighting that user and clicking on the left arrow. The user moves from the **Members Selected** to **Members Available** list, and no longer has the permissions associated with the highlighted role.

The users that appear in the **Members Selected** list are now tied to the highlighted role, and have all of the permissions associated with that role.

Set Up an Email Server

A convenient way to share reports is to set up an email server that can send reports to recipients. This feature works with the report scheduling feature to automate the process of emailing reports to your user community. Setting up an email server is not required. If you want to get started quickly or do not have information about your email server, skip this for now and go on to the next stop on the *Guide Post* graphic. You can always come back to it later.

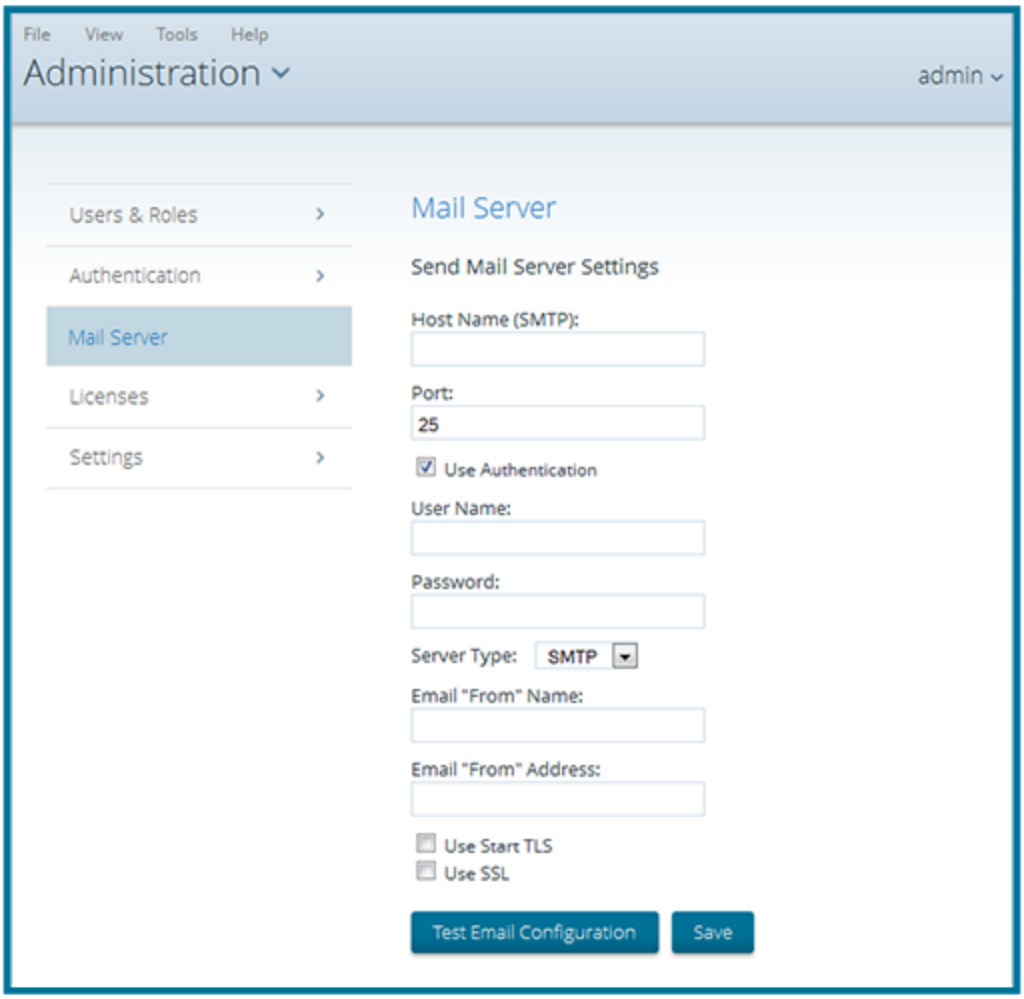
If you want to set up the email server, here are the tasks you must perform.

- Tour the Mail Server Page
- Configure Email Notification

When you are done, please go on to the next stop on the *Guide Post* graphic.

Tour the Mail Server Page

You use the **Administration** page within the User Console and access **Mail Server** to set up the e-mail server.



Setting	Description
Host Name (SMTP)	Address of your SMTP email server for sending email.
Port	Port of your SMTP email server, usually 25. For Gmail, the value is 587.
Use Authentication	Enable to use authentication for email.
User Name	User ID to connect to the email server for sending email.
Password	Password used to connect to the email server.
Server Type	Transport for accessing the email server, usually SMTP. For Gmail, SMTP is required.
Email "From" Name	Name that appears in the **From** field in an email. If left blank, the default email name for this field is Pentaho Scheduler.
Email "From" Address	Address that appears in the **From** field in an email. If left blank, the default email address for this field is Pentaho Scheduler.
Use Start TLS	Enable if the email server requires a Start TLS connection.

Setting	Description
Use SSL	Enable if the email server requires an SSL connection. This value must be enabled for Gmail.

Configure Email Notification

1. Log into the User Console, click **Administration** in the upper-right corner, then click **Mail Server** from the items on the left. The **Mail Server** page appears.
2. Enter your email server settings.
3. Click **Test Email Configuration**, then click **Save**. A success message appears.

Next Steps

There are only a few last things you need to do, and then you are done configuring the BA Server and User Console.

1. Document how you have configured the BA Server. When you get a newer version of Pentaho, this helps you decide which components need upgrading and how to configure the upgraded components if necessary.
2. Share the BA Server URL with team members who want to create web-based data analysis, reports, and dashboards. This enables them to login to the User Console and get started with the Web-Based Analysis, Reports, and Dashboards Tutorial.
3. If you want to use the BA design tools, please go on to configure them *now*. This way they will be ready to use when you need them.

Learn More

- The *Administer Pentaho Systems* section of the Pentaho InfoCenter contains advanced administration topics. Here you can learn about refining your configuration, fine-tuning system performance, or troubleshoot the BA Server and User Console.
- Build Web-Based Data Analysis, Reports, and Dashboards provides in-depth details about creating eye-catching business intelligence deliverables for your user community.
- Work with Big Data tells you how to use big data as a data source.

Configure the BA Design Tools

Before you can use the BA design tools, you need to perform configuration tasks for each workstation that runs the design tools. Just follow the *Guide Post* graphics to ensure you complete the entire process. It should take you less than 1 hour for each workstation running the design tools.

If you used the **Pentaho Trial Download** and the Installation Wizard, the BA Server and its design tools reside on the same workstation. If you used any of the other installation techniques, the BA Server and design tools may reside on separate workstations. Here is the list of the design tools and what they do.

Design Tool	What It Does
Aggregation Designer	Optimizes the multidimensional Mondrian data model
Metadata Editor	Creates relational data models and refines the models created with the Data Access Wizard
Report Designer	Creates print-quality reports
Schema Workbench	Creates multidimensional Mondrian data models and refines the models created with the Data Access Wizard

Start and Stop BA Design Tools

Each of the BA design tools has a specific function, so you start and stop each one individually. As part of the Installation Wizard, Archive Installation, and Manual Installation, we provide different ways to help you start each design tool, depending on the operating system you use. This table shows you options. After you have decided how you want to start and stop the design tools, please go on to the next stop on the *Guide Post* graphic.

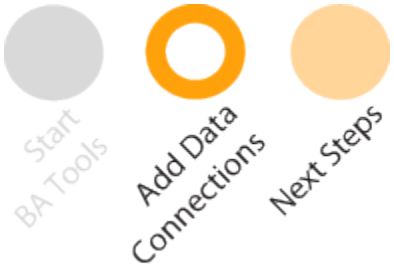
Operating System	Installation Method	Start the Design Tools
Microsoft Windows	Pentaho Installation Wizard	Click **Start** > **Programs** > **Pentaho Enterprise Edition** > **Design Tools** and double-click one of these items to launch the design tool. • Aggregation Designer • Metadata Editor • Report Designer • Schema Workbench
Microsoft Windows	Pentaho Archive or Manual Installation	Create shortcuts to the `.bat` file or each design tool and add them to the Start menu. Or, navigate to the one of these directories and run the design tool's `.bat` file to launch it. Aggregation Designer • `\pentaho\design-tools\aggregation-designer\startaggregationdesigner.bat` Metadata Editor • `\pentaho\design-tools\metadata-editor\metadata-editor.bat` Report Designer • `\pentaho\design-tools\report-designer\report-designer.bat` Schema Workbench

Operating System	Installation Method	Start the Design Tools
		• `\pentaho\design-tools\schema-workbench\workbench.bat`
Linux	Pentaho Wizard, Archive, or Manual Installation	Navigate to the one of these directories and run the design tool's `.sh` file to launch it. Aggregation Designer • `/pentaho/design-tools/aggregation-designer/startaggregationdesigner.sh` Metadata Editor • `/pentaho/design-tools/metadata-editor/metadata-editor.sh` Report Designer • `/pentaho/design-tools/report-designer/report-designer.sh` Schema Workbench • `/pentaho/design-tools/schema-workbench/workbench.sh`
Mac OS	Pentaho Wizard, Archive, or Manual Installation	Navigate to the one of these directories and double-click the `.app` file to launch the design tool. • `/pentaho/design-tools/aggregation-designer/Aggregation Designer.app` • `/pentaho/design-tools/metadata-editor/Metadata Editor 32-bit.app` or `Metadata Editor 64-bit.app` • `/pentaho/design-tools/Pentaho Report Designer.app` • `/pentaho/design-tools/schema-workbench/Schema_Workbench.app`

Stop the Aggregation Designer by clicking on the **x** icon in upper corner.

Stop the Metadata Editor, Report Designer, and Schema Workbench from the **File** menu.

Specify Data Connections for BA Design Tools



The BA design tools need a way to connect to where you store data, as well as to the BA Server where you publish and share what you create in the design tools. We support connecting to data stored in these ways.

- Pentaho data sources, such as relational Metadata data models or multi-dimensional Mondrian data models
- JDBC databases

- **Define JDBC or OCI Connections For BA Design Tools**
- **Define JNDI Connections for Report Designer and Metadata Editor**

JDBC Database Connections

How you connect to databases depends on which access protocol is best to use when creating reports and data models or publishing them. You can choose from these access protocols.

- Native (JDBC)—This is a commonly used access protocol. Please see details in the Database Connection Access Type Decision Table.
- JNDI—This also is a commonly used access protocol. Please see details in the Database Connection Access Type Decision Table.
- ODBC—We do not support ODBC, but we make it available in case you need it. If you must use ODBC, contact Pentaho support to ensure you are successful.
- OCI—If you are connecting to an Oracle database, skip the rest and go on to define the OCI connection.

Table 1. Database Connection Access Type Decision Table

Explore Considerations	Choose Options	
	Native (JDBC)	JNDI
Summary	These connections are the easiest to define. However, they do not scale well and are not easy to maintain. If the connection information changes, you must change it in every unpublished or published data	These connections are easier to maintain since only the connection name is stored, if the connection properties are stored in a file on each workstation the runs design tools.

Explore Considerations	Choose Options	
	Native (JDBC)	JNDI
	model or report you have created with the design tools.	
	Connection Behavior for Report Designer and Metadata Editor The connection information is embedded into each report or Metadata data model. Once you have defined connections, the Report Designer displays the connection name in a list where you can select it. However, for Metadata Editor, once you have defined connections, you must enter the right connection name each time you use the editor. *Before* you publish, ensure that the connection name is same name as a Native (JDBC) connection already defined on the BA server.	**Connection Behavior for Report Designer and Metadata Editor** The connection information is stored in a properties file on each workstation that runs these tools. *Before* you publish, ensure that the JNDI connection has the same name as one already defined on the BA server.
	Connection Behavior for Schema Workbench and Aggregation Designer Connection information is not embedded or stored. You must define the connection each time you use these tools. *Before* you publish, you need to know the URL of the BA Server and the name of a Native (JDBC) connection already defined on that server.	**Connection Behavior for Schema Workbench and Aggregation Designer** JNDI is not an option for these tools. *Before* you publish, you need to know the URL of the BA Server and the name of a JNDI connection already defined on that server.
Expertise	Knowledge of the JDBC driver and options for your RDBMS	
Time	Approximately 10 minutes for each workstation running the design tools	Approximately 30 minutes for each for each workstation running the design tools
Recommendation	**Use for the Pentaho Trial Download and evaluation. But, switch each report and data model to JNDI *before* publishing to avoid problems if connection information changes.**	**Use for production or when the work environment is distributed in a network.**

Define JDBC or OCI Connections For BA Design Tools

Once you have chosen to use a JDBC or OCI access protocol, all you have to do *now* is install the appropriate driver so that you can use dsign tools anytime you want. You will specify the access protocol later when you define the data source for each design tool.

When the driver file is in place, go on to the next stop on the *Guide Post* graphic.

Add Drivers

The driver enables design tools to connect to the BA Server and verify that the model is correct. Your database administrator, Chief Intelligence Officer, or IT manager should be able to provide the appropriate driver. If not, you can download drivers from your database vendor's website. See the Supported Technologies to ensure that your database and its driver is supported.

When you have the correct driver, copy it to these directories on *all* machines that run the design tools you chose to install. Design tools should not be running when you do this. Once the driver is in place, you can start the design tools.

- **Aggregation Designer:** `/pentaho/design-tools/agg-designer/drivers/`
- **Metadata Editor:** `/pentaho/design-tools/metadata-editor/libext/JDBC/`
- **Report Designer:** `/pentaho/design-tools/report-designer/lib/jdbc/`
- **Schema Workbench:** `/pentaho/design-tools/schema-workbench/drivers/`

There should be only one driver for your database in this directory. Ensure that there are no other versions of the same vendor's driver in this directory. If there are, back up the old driver files and remove them to avoid version conflicts. If you have any concerns about how to proceed, contact Pentaho support.

Driver for Microsoft SQL Server

If you are using a Microsoft SQL Server (MSSQL), you might need to use an alternative, non-vendor-supported driver called JTDS. Contact Pentaho support to ensure that you are adding the correct driver.

For Microsoft Windows, most JDBC drivers support Type 2 integrated authentication through the `integratedSecurity` connection string property. To use integrated authentication, copy the `sqljdbc_auth.dll` file to all machines and directories to which you copied the JDBC driver. You can find this file in this location.

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

- 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.

Define JNDI Connections for Report Designer and Metadata Editor

Once you have chosen to use the JNDI access protocol, here are configuration tasks you can perform.

- Add Drivers
- Edit the Properties File for Report Designer
- Edit the Properties File for Metadata Editor

When you are done, please go on to the next stop on the *Guide Post* graphic.

Add Drivers

The driver enables design tools to connect to the BA Server and verify that the model is correct. Your database administrator, Chief Intelligence Officer, or IT manager should be able to provide the appropriate driver. If not, you can download drivers from your database vendor's website. See the Supported Technologies to ensure that your database and its driver is supported.

When you have the correct driver, copy it to these directories on *all* machines that run the design tools you chose to install. Design tools should not be running when you do this. Once the driver is in place, you can start the design tools.

- **Aggregation Designer:** `/pentaho/design-tools/agg-designer/drivers/`
- **Metadata Editor:** `/pentaho/design-tools/metadata-editor/libext/JDBC/`
- **Report Designer:** `/pentaho/design-tools/report-designer/lib/jdbc/`
- **Schema Workbench:** `/pentaho/design-tools/schema-workbench/drivers/`

There should be only one driver for your database in this directory. Ensure that there are no other versions of the same vendor's driver in this directory. If there are, back up the old driver files and remove them to avoid version conflicts. If you have any concerns about how to proceed, contact Pentaho support.

Driver for Microsoft SQL Server

If you are using a Microsoft SQL Server (MSSQL), you might need to use an alternative, non-vendor-supported driver called JTDS. Contact Pentaho support to ensure that you are adding the correct driver.

For Microsoft Windows, most JDBC drivers support Type 2 integrated authentication through the `integratedSecurity` connection string property. To use integrated authentication, copy the `sqljdbc_auth.dll` file to all machines and directories to which you copied the JDBC driver. You can find this file in this location.

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

- 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.

Edit the Properties File for Report Designer

1. On the workstation where you want to run Report Designer, stop Report Designer and the BA server if it is running on the same workstation.
2. Navigate to the `.pentaho` directory in the user or home directory. For example, if the user name is `fbueller` for Microsoft Windows, the pentaho directory is `C:\Users\fbueller\.pentaho\`, and for Linux or Solaris the directory is `/home/fbueller/.pentaho/`.
3. Switch to the `~/.pentaho/simple-jndi/` subdirectory. If it does not exist, create it.
4. Edit the `default.properties` file found there. If it does not exist, create it according to this example.

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

   In the example, `SampleData` is the name of the JNDI connection. Each line must begin with the JNDI connection name and a forward slash (/), followed by these required parameters.

Parameter	Description
type	`javax.sql.DataSource` defines a JNDI data source type.
driver	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.

5. Save and close the file.

6. In order for this change to take effect,restart the design tools and the server if the server and design tools are running on the same workstation.

You now have a properties file that defines a JNDI connection for Report Designer. Remember, if you run this tool on more than one workstation, repeat this process on *all* of the other workstations.

Edit the Properties File for Metadata Editor

1. On the workstation where you want to run Metadata Editor, stop Metadata Editor and the server if it is running on the same workstation.
2. Navigate to the `metadata-editor/simple_jndi` directory where you installed the Metadata Editor.
3. Edit the `JDBC.properties` file found there. If it does not exist, create it according to this example.

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

   In the example, `SampleData` is the name of the JNDI connection. Each line must begin with the JNDI connection name and a forward slash (/), followed by these required parameters.

Parameter	Description
type	`javax.sql.DataSource` defines a JNDI data source type.
driver	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.

4. Save and close the file.
5. For these changes to take effect, restart the Metadata Editor and the BA server if it is running on the same workstation.

You now have a properties file that defines a JNDI connection for Report Designer. Remember, if you run this tool on more than one workstation, repeat this process on *all* the other workstations.

Next Steps

There are only a few last things you need to do, and then you are done configuring the Pentaho design tools.

1. Document how you have configured the tools. When you get a newer version of Pentaho, this helps you decide which tools need upgrading and how to configure the upgraded tools if necessary.
2. Share how to start the tools with team members who may need to use them. Here is the documentation for the design tools.
    - Print-Quality Reports Tutorial using the Report Designer
    - Design Print-Quality Reports using the Report Designer
    - Work with Relational Metadata Data Models using Metadata Editor
    - Work with Mondrian Multi-Dimensional Data Models using Schema Workbench
    - Optimize Mondrian Data Models using Aggregation Designer
3. If you want to use the DI Server and Spoon for ETL processes, please go on to configure them now. This way they will be ready to use when you need them.

Learn More

Work with Big Data tells you how to use big data as a data source.

The topics at the end of these sections contain administration instructions for the design tools.

- Design Print-Quality Reports
- Work with Relational Metadata Data Models
- Work with Mondrian Multi-Dimensional Data Models
- Optimize Mondrian Data Models