Pentaho BI Suite Mac Installation 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

Mac Installation Overview

Pentaho provides several installation paths to meet a variety of customer needs. This guide represents the simplest procedure for installing the latest release of the Pentaho BI Suite Enterprise Edition on Apple Mac machines.

The Pentaho Mac Archive Packages Will Provide	You Must Supply On Your Own
Application server: Tomcat 6.0 **BI Server:** • Pentaho User Console • Pentaho Analyzer • Pentaho Dashboard Designer • Pentaho Enterprise Console **Data preparation tools:** • Schema Workbench • Pentaho Data Integration **Design tools:** • Report Designer • Design Studio	**A Mac with OS X version 10.5 or newer** **A database to store solutions in:** • MySQL 5.0 • PostgreSQL 8.2 • Oracle 10g **One or more data sources:** • Any JDBC-compliant database • A spreadsheet • A flat file containing comma-separated values

Are you installing on Linux, Solaris, or Windows? This guide only covers OS X. If you are using another operating system on a Mac or any other machine, refer instead to the *Pentaho BI Suite Archive-Based Installation Guide*.

If you would prefer to build the application from source, if you want to use you own extant Tomcat or JBoss application server, or if you plan to implement single sign-on through Central Authentication Service, refer instead to the *Pentaho BI Server Manual Deployment Guide*.

Installation Checklist

The Installation Checklist is a concise list of instructions intended to show a high-level overview of the BI Suite installation and configuration process for Mac computers. It also serves as a quick reference for administrators and developers who have performed several installations in the past and only need a brief rehash of the necessary steps. **This is not the complete instruction set.** If you need more details than are provided in this checklist, consult the appropriate section in the verbose instruction set that comprises the rest of this guide.

Step	Procedure	Done
Step 1	Download the BI Server and Dashboard Designer archive packages from the Enterprise Edition FTP site.	
Step 2	Create an /Applications/pentaho/server/ directory, then unpack the BI Server archive into it.	
Step 3	Using the included SQL scripts for your database of choice, initialize your Pentaho solution repository.	
Step 4	For Oracle or PostgreSQL solution databases, modify the **server.xml**, **pentaho.xml**, **hibernate-settings.xml**, and **applicationContext-spring-security-hibernate.properties** files appropriately, and copy or modify the audit logging materials.	
Step 5	If necessary, modify the appropriate XML configuration files to change the default hostname from **localhost**.	
Step 6	If necessary, modify Tomcat's **server.xml** to change the BI Server and Enterprise Console port numbers.	
Step 7	If you have a Dashboards Enterprise Edition, Pentaho Analysis Enterprise Edition, and/ or Pentaho BI Suite For Hadoop support entitlement, install the Pentaho Dashboard Designer, Pentaho Analyzer, and Pentaho Hadoop Distribution plugins by unpacking them to the proper locations in `/pentaho/server/biserver-ee/pentaho-solutions/system/`.	
Step 8	Start the Pentaho BI Server and Pentaho Enterprise Console.	
Step 9	Access the Pentaho Enterprise Console at `http://localhost:8088` to install Enterprise Edition license keys, and establish users, roles, data sources, and any other configuration tasks. Use the *Pentaho BI Suite Administrator's Guide* and the *Pentaho BI Suite Security Guide* for assistance.	
Step 10	Optionally, modify your OS X startup configuration so that the BI Platform and Pentaho Enterprise Console will start automatically at boot time.	
Step 11	Download the archive packages for all Pentaho client tools that you have support entitlements for, and unpack them on each workstation.	
Step 12	Access the BI Server from a workstation and verify that you can log in as a normal user, and create reports and analysis views.	

Prerequisites

In order to install the BI Server on a Mac, you must know your server's administrator password, or have the cooperation of someone who does. Desktop machines that will only run the Pentaho client tools do not require any extra access privileges.

Note: Usually you do not need to do much work to install software on a Mac. However, the Pentaho BI Suite does not yet have a Mac installation utility, so the installation process requires low-level system configuration work that many Mac users are not accustomed to. This guide is intended for system administrators and other technologically confident professionals; end-users may have difficulty following some of the more sophisticated procedures below.

Apple Hardware Requirements

Pentaho Enterprise Edition software does not have strict limits on computer or network hardware. As long as you meet the minimum software requirements (note that your operating system will have its own minimum hardware requirements), Pentaho is hardware agnostic. There is, however, a recommended set of system specifications for servers and high-end production workstations:

RAM	At least 2GB
Hard drive space	At least 2GB
Model	XServe

It's possible to use a less capable system, but in most realistic scenarios, the too-limited system resources will result in an undesirable level of performance. Theoretically you can use any Mac running any operating system that supports the Sun JRE version 1.6 (6.0).

Your software environment does not have to be 64-bit, even if your processor architecture supports it.

Software Requirements

In terms of operating systems, Mac OS X 10.5 or newer is supported. If you are using another supported operating system on Mac hardware, refer to one of the other installation guides instead.

No matter which operating system you use, you must have the Sun Java Runtime Environment (JRE) version 1.6 (6.0) installed. Other versions will not work with all Pentaho servers and client tools.

Mac desktops will need to have reasonably modern Web browsers to access Pentaho's Web interface. Firefox 3.5 or higher; and Safari 4.0 or higher will both work.

Your environment can be either 32-bit or 64-bit as long as it meets the above requirements.

You must have one of the following databases installed, configured, and accessible from the machine that you're installing the BI Server on: Oracle 10g, MySQL 5.0, or PostgreSQL 8.2.

The aforementioned configurations are officially supported by Pentaho. Other operating systems such as FreeBSD and OpenBSD; other Java virtual machines like Blackdown; and other Web browsers like Opera might work without any problems. However, the Pentaho support team may not be able to help you if you have trouble installing or using the BI Suite under these conditions.

Supported Solution Databases

You must have a supported database installed, running, and available before you continue. The database can be running on the same server that the BI Server is being installed to, or it can be on a remote machine that is accessible from this one. The BI Server 3.8 can use the following databases to store the solutions:

- MySQL version 5.0 or higher
- PostgreSQL version 8.2 or higher
- Oracle 10g

Obtaining the OS X Installation Materials

Enterprise Edition customers can obtain the BI Server and client tool pre-configured packages from the Pentaho Enterprise Edition FTP site by using their Pentaho account login credentials. If you are unfamiliar with these details, consult the Welcome Kit provided to you by Pentaho customer support as part of the Enterprise Edition enablement process.

The **Pentaho User Console plugins** item below refers to add-ons for the Pentaho User Console, such as Dashboard Designer and Pentaho Analyzer.

The packages you need for your server are:

- **BI Server (includes the Pentaho Enterprise Console):** /3.8.0-GA/server/nix/biserver-ee-3.8.0-GA.tar.gz
- **Pentaho User Console plugins:** /3.8.0-GA/server/plugins/
- **BI Server Hadoop plugin:** /Pentaho_Data_Integration/4.1.2-GA/server/plugins/pdi-hadoop-plugin-1.0.3.zip
- **BI Server Hadoop Enterprise Edition plugin:** /Pentaho_Data_Integration/4.1.2-GA/server/plugins/pdi-hadoop-plugin-ee-1.0.4.zip

The packages you will need for your workstations are:

- **Report Designer:** /3.8.0-GA/client/nix/prd-ee-mac-3.8.0-GA.tar.gz
- **Metadata Editor:** /3.8.0-GA/client/nix/pme-ee-3.8.0-GA.tar.gz
- **Schema Workbench 32-bit:** /3.8.0-GA/client/nix/psw-ee-3.2.0.13885.tar.gz
- **32-bit Design Studio:** /3.8.0-GA/client/nix/pds-ee-mac-32-3.8.0-GA.tar.gz
- **64-bit Design Studio:** /3.8.0-GA/client/nix/pds-ee-mac-64-3.8.0-GA.tar.gz
- **Aggregation Designer:** /3.8.0-GA/client/nix/pad-ee-1.2.1-GA.tar.gz

Note: The version numbers in the paths and filenames above may change due to minor point releases, which do not generally necessitate documentation updates.

With the exception of Design Studio, which has separate packages for 32-bit and 64-bit Linux, OS X, and Windows, all of the above-listed files are platform-agnostic, so there is only one download for all operating systems, databases, and application servers. Most windows servers do not typically have software to unpack a .tar.gz file, so it is repackaged in .zip format for Windows users. Other than this archive technology difference, the BI Server packages are identical.

Deploying the BI Server

Create a directory in your Applications folder called `/pentaho/server/`, then untar the BI Server package into it.

👉 **Note:** All further instructions in this guide will assume that you have established this directory structure, and will refer to it as **/pentaho/server/**. If you choose a different directory name or path, you will have to adjust the instructions accordingly.

Once you've unpacked the archive, you must configure both the BI Server and the included Tomcat application server for your repository database. This involves initializing your database, then adding the appropriate database drivers for your application server. Below are instructions for all supported databases. Only follow the sections that apply to your software environment.

Initializing a MySQL Database

In order to complete this process, you need to have the MySQL client installed on your machine, and the MySQL database server must be running as a service.

⚠️ **Caution:** You must use the **ASCII** character set in order for these scripts to execute properly. UTF-8 is known not to work because of text string length limitations.

This procedure will initialize Pentaho-related databases on your MySQL server. You must execute the scripts below in the order they are presented in the instructions, or else the solution repository database may not be properly initialized.

1. Navigate to the `/pentaho/server/biserver-ee/data/mysql5/` directory.

    There are several scripts in this directory. You should only run the ones mentioned below.
2. Execute the **create_quartz_mysql.sql** script via your preferred utility or process.

    There are a variety of ways to execute an SQL script. The simplest way is by using the MySQL client from the terminal.
    ```
    mysql -u root -p <create_quartz_mysql.sql
    ```
3. Execute the **create_repository_mysql.sql** script.
    ```
    mysql -u root -p <create_repository_mysql.sql
    ```

MySQL is now configured to act as a Pentaho BI Server solution repository.

Initializing a PostgreSQL Database

In order to complete this process, you need to have the PostgreSQL native client tools installed, and the database server running as a service. Your PostgreSQL configuration must also support logins from all users; this is not the default configuration in all cases, so you may have to edit your **pg_hba.conf** file to support this option.

This procedure will initialize Pentaho-related databases on your PostgreSQL server.

1. Navigate to the `/pentaho/server/biserver-ee/data/postgresql/` directory.

    There are several scripts in this directory. You should only run the ones mentioned below.
2. Edit the **create_repository_postgresql.sql** and **create_quartz_postgresql.sql** scripts and set the password references to the passwords you want to use for the Pentaho administrator and user accounts.

    The user account specified here is not for a specific Pentaho BI Server user; rather it is a generic account that the software uses to update the database when any regular user is logged in.
3. Execute the schedule database initialization script as the PostgreSQL root user by running this command from a terminal (change "postgres" to the name of your PostgreSQL root user): **psql -U postgres < create_quartz_postgresql.sql**.
4. Next, initialize your repository database: **psql -U postgres < create_repository_postgresql.sql**.

PostgreSQL is now configured to act as a Pentaho BI Server solution repository.

PostgreSQL Solution Repository Configuration

Follow the below directions to configure the BI Server to use PostgreSQL as a solution repository with the default Pentaho security data access object. Even if you will end up using some other authentication method, follow all of these directions anyway so that you can verify that the BI Server works with a basic configuration.

1. Open the `/pentaho/server/biserver-ee/pentaho-solutions/system/applicationContext-spring-security-hibernate.properties` file in a text editor.
2. Edit the file and change the following values to those in this example, modifying the details for your specific PostgreSQL configuration:

   👉 **Note:** Alternatively, you can copy over a PostegreSQL-primed version of this file from /pentaho-solutions/system/dialects/postgresql/ and then change the details by hand.

   ```
   jdbc.driver=org.postgresql.Driver
   jdbc.url=jdbc:postgresql://localhost:5432/hibernate
   jdbc.username=admin
   jdbc.password=pentaho
   hibernate.dialect=org.hibernate.dialect.PostgreSQLDialect
   ```

3. Save and close the file, then edit `/pentaho/server/biserver-ee/tomcat/webapps/pentaho/META-INF/context.xml`
4. Change the hibernate and quartz **driverClassName** entries to `org.postgresql.Driver`
5. Change the hibernate **url** property to:

   ```
   jdbc:postgresql://localhost:5432/hibernate
   ```

   👉 **Note:** If you are using a different port number, change it accordingly in this and the next step. If your PostgreSQL database server is not hosted on the same machine as your BI Server, you will need to change **localhost** to the hostname or IP address of your PostgreSQL database server. There are other files you must modify as well; refer to the Changing the Solution Database Hostname later in this guide for further instruction.

6. Change the quartz **url** property to `jdbc:postgresql://localhost:5432/quartz`
7. Replace the value of **validationQuery** with **"select 1"**.

   ```
   validationQuery="select 1"
   ```

8. Save and close the file, then edit `/pentaho/server/biserver-ee/pentaho-solutions/system/hibernate/hibernate-settings.xml`
9. Replace the value of the <config-file> element with the following:

   ```
   system/hibernate/postgresql.hibernate.cfg.xml
   ```

10. Save and close the file, then edit the database-specific configuration file that you just specified in the config-file element.
11. Change the default hostname, port number, and user credentials to match your database configuration, then save and close the file.

    Even if you plan to use the default settings, open the file to verify that the defaults are what you expect them to be.

The BI Server is now fully configured to connect to a PostgreSQL solution repository.

Configuring Quartz For PostgreSQL

The default JobStore JDBC database driver does not work properly with PostgreSQL. Follow the below procedure to replace the generic JDBC driver with one specific to PostgreSQL.

1. Open the `/pentaho/server/biserver-ee/pentaho-solutions/system/quartz/quartz.properties` file in a text editor.
2. Find the following line in the properties file: `org.quartz.jobStore.driverDelegateClass = org.quartz.impl.jdbcjobstore.StdJDBCDelegate`
3. In the aforementioned line, replace the **org.quartz.impl.jdbcjobstore.StdJDBCDelegate** portion with this: **org.quartz.impl.jdbcjobstore.PostgreSQLDelegate**
4. Save the file and close the text editor.

The BI Server's scheduling (quartz) database is now configured to work with PostgreSQL.

Configuring Audit Logging For PostgreSQL

Audit logging is enabled by default in the BI Server, but it expects a MySQL database. To configure it for PostgreSQL, follow the below process.

1. Copy all of the files from `/pentaho/server/biserver-ee/pentaho-solutions/admin/audit/dialects/postgresql/` to the `/pentaho/server/biserver-ee/pentaho-solutions/admin/audit/` directory, overwriting the existing files there.
2. Copy the **audit_sql.xml** file from `/pentaho/server/biserver-ee/pentaho-solutions/system/dialects/postgresql/` to `/pentaho/server/biserver-ee/pentaho-solutions/system/`, overwriting the existing version there.

Audit logging is now configured for PostgreSQL and working as intended.

Initializing an Oracle Database

In order to complete this process, Oracle should be installed and running, and you should have a basic familiarity with Oracle database management.

This procedure will initialize a Pentaho table space on your Oracle server.

1. Start the Oracle Database Configuration Manager and use it to create a database called **pentaho**.

   This database will house your BI Server solutions.
2. Navigate to the `/pentaho/server/biserver-ee/data/oracle10g/` directory.

   There are several scripts in this directory. You should only run the ones mentioned below.
3. Edit the **create_quartz_ora.sql** and **create_repository_ora.sql** scripts and replace the **admin/password** portion of the **conn** lines with the username and password you use to manage your Oracle database.

   The user credentials you type in here should have permission to create tables in the **pentaho** database, create users to access that database, and create and manage permissions for tables in it as well. This connection instance will be called **sampledata**.
4. Edit the **datafile** path with the path to your Oracle installation, then save and close your text editor.
5. Run the SQL scripts with your preferred utility or process, in the order shown above.

   SQL*Plus is probably the easiest tool to use for this purpose.

Oracle is now configured to act as a Pentaho BI Server data repository.

Oracle Solution Repository Configuration

Follow the below directions to configure the BI Server to use Oracle as a solution repository with the default Pentaho authentication backend. Even if you will end up using some other authentication method, follow all of these directions anyway so that you can verify that the BI Server works with a basic configuration.

1. Open the `/pentaho/server/biserver-ee/pentaho-solutions/system/applicationContext-spring-security-hibernate.properties` file in a text editor.
2. Edit the file and change the following values to those in this example, modifying the details for your specific Oracle configuration:

```
jdbc.driver=oracle.jdbc.driver.OracleDriver
jdbc.url=jdbc:oracle:thin:@localhost:1521:XE
jdbc.username=hibuser
jdbc.password=password
hibernate.dialect=org.hibernate.dialect.Oracle10gDialect
```
3. Save and close the file, then edit `/pentaho/server/biserver-ee/pentaho-solutions/system/hibernate/hibernate-settings.xml`
4. Replace the value of the <config-file> element with the following:

   ```
   system/hibernate/oracle10g.hibernate.cfg.xml
   ```
5. Save and close the file, then edit `/pentaho/server/biserver-ee/tomcat/webapps/pentaho/META-INF/context.xml`
6. Change the hibernate and quartz **driverClassName** entries to `oracle.jdbc.OracleDriver`

7. Change the hibernate **url** property to `jdbc:oracle:thin:@localhost:1521:XE`

   If you are using a different port number, change it accordingly in this and the next step. If your Oracle database server is not hosted on the same machine as your BI Server, you will need to change **localhost** to the hostname or IP address of your Oracle machine. Lastly, **XE** must change to match your Oracle service name. There are other files you must modify as well; refer to the Changing the Solution Database Hostname later in this guide for further instruction.

8. Change the quartz **url** property to the following:

   ```
   jdbc:oracle:thin:@localhost:1521:XE
   ```

9. Save and close the file, then edit the database-specific configuration file that you just specified in the config-file element.

10. Change the default hostname, port number, and user credentials to match your database configuration, then save and close the file.

   Even if you plan to use the default settings, open the file to verify that the defaults are what you expect them to be.

The BI Server is now fully configured to connect to an Oracle solution repository.

Configuring Audit Logging For Oracle

Audit logging is enabled by default in the BI Server, but it expects a MySQL database. To configure it for Oracle, follow the below process.

Copy all of the files from `/pentaho/server/biserver-ee/pentaho-solutions/admin/audit/dialects/oracle10g/` to the `/pentaho/server/biserver-ee/pentaho-solutions/admin/audit/` directory, overwriting the existing files there.

Audit logging is now configured for Oracle and working as intended.

Changing the Solution Database Hostname

The Pentaho BI Server's default configuration assumes that your solution database server and the Pentaho BI Server are going to be on the same machine, both accessible via **localhost**. If they are not on the same machine, you will have to modify some files in order to connect the two servers.

1. Navigate to your newly deployed WAR or EAR file.
2. For EAR deployments, edit the `PentahoHibernate-ds.xml` and `quartz-ds.xml` files in the EAR root directory, and replace the **localhost** reference in each with the hostname, domain name, or IP address of your solution database server.
3. For WAR deployments, navigate to the `/META-INF/` directory in your pentaho.war file, and edit the `context.xml` file, replacing the **localhost** reference with your solution database server's hostname, domain name, or IP address.
4. Navigate to the `/pentaho/server/biserver-ee/pentaho-solutions/system/` directory.
5. Change the **localhost** reference appropriately in the `applicationContext-spring-security-hibernate.properties` file.
6. Change to the `hibernate` subdirectory.
7. Edit the `database.hibernate.cfg.xml` file, where *database* is the name of your solution database (shown below) and change the **localhost** reference appropriately.

   The file you have to edit will be one of these:

   * hsql.hibernate.cfg.xml
   * mysql5.hibernate.cfg.xml
   * postgresql.hibernate.cfg.xml
   * oracle10g.hibernate.cfg.xml

The Pentaho BI Server is now configured to connect to your remote solution database.

Changing the Default Port Numbers

The Pentaho-provided Tomcat instance is preconfigured to run on port **8080**, and the Pentaho Enterprise Console is configured for port **8088**. If you want to change these port numbers, follow the below procedure.

1. Navigate to the `/pentaho/server/biserver-ee/tomcat/conf/` directory

2. Edit the **server.xml** file found there, and search for **Define a non-SSL HTTP/1.1 Connector**. Change the port number in the **Connector port** element below that from **8080** to your preferred port number.

```
<!-- Define a non-SSL HTTP/1.1 Connector on port 8080 -->
    <Connector port="8080" maxHttpHeaderSize="8192"
               maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
               enableLookups="false" redirectPort="8443" acceptCount="100"
               connectionTimeout="20000" disableUploadTimeout="true" />
```

3. Save and close the file, then navigate to the `/pentaho/server/biserver-ee/tomcat/webapps/pentaho/WEB-INF/` directory and edit the **web.xml** file found there.

4. Change the **fully-qualified-server-url** entry to match the new port number you specified in server.xml.

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

5. Save and close the file, then navigate to the `/pentaho/server/enterprise-console/resource/config/` and edit the **console.properties** file found there.

6. Change the **console.start.port.number** and **console.stop.port.number** values to match the port numbers you want to run Enterprise Console on.

```
console.start.port.number=8088
console.stop.port.number=8033
```

Installing Pentaho User Console Plugins

In order to proceed with this task, you must have already purchased an Enterprise Edition support entitlement for the products you want to install. Skip this process if you do not have the requisite support entitlements.

The Pentaho User Console is built with a plugin architecture that enables you to expand its functionality with new client tools and functions. Follow the directions below to install any of the following plugins:

- Pentaho Analyzer
- Pentaho Dashboard Designer

> **Note:** Pentaho Analyzer requires that Dashboard Designer also be installed.

1. If you have not already done so, download the plugin packages from the Pentaho Enterprise Edition FTP site, or the Pentaho Knowledge Base.

   The Dashboard Designer package name always begins with **pdd**; the Pentaho Analyzer package always begins with **paz**.

2. Unpack the plugin packages to the `/pentaho/server/biserver-ee/pentaho-solutions/system/` directory.

   This will create a subdirectory for the plugin.

The plugins you downloaded are now installed, though you will still need to register your license files in the Pentaho Enterprise Console, or with the command line license management tool. There are instructions for that process later in this guide. For now, continue on to the next section.

Installing Hadoop Hive Plugins

In order to proceed with this task, you must have already purchased a Pentaho BI Suite For Hadoop support entitlement. Skip this process if you do not have one.

The BI Server requires special Pentaho Data Integration plugins in order to use Hive metadata, and to use PDI transformations in action sequences. Follow the instructions below to install these plugins in the BI Server.

1. If you have not already done so, download the plugin packages from the Pentaho Enterprise Edition FTP site, or the Pentaho Knowledge Base.

   The package names are:

   - pdi-hadoop-plugin
   - pdi-hadoop-plugin-ee

2. Unpack the plugin packages to the `/pentaho/server/biserver-ee/pentaho-solutions/system/kettle/plugins/` directory.

The plugins you downloaded are now installed, though you will still need to register your license file in the Pentaho Enterprise Console, or with the command line license management tool. There are instructions for that process later in this guide. For now, continue on to the next section.

Initial Startup of the BI Server and Pentaho Enterprise Console

The basic installation procedures are now complete, so it's time to start the BI Server and Pentaho Enterprise Console so that you can proceed with low-level system configuration.

1. Execute the `/pentaho/server/biserver-ee/start-pentaho` script.

   This script has a **.bat** extension on Windows, and a **.sh** extension on Linux, Solaris, and OS X.

2. Execute the `/pentaho/server/enterprise-console/start-pec` script.

   This script has a **.bat** extension on Windows, and a **.sh** extension on Linux, Solaris, and OS X. It may take a few moments to compete the startup sequence.

Both the Pentaho BI Server and the Pentaho Enterprise Console server are now running and should be accessible via a Web browser.

Post-Install Configuration

After you've successfully built and deployed the BI Server, you must set up your configuration information and other details. All of this is done through the Pentaho Enterprise Console.

1. Open a Web browser and navigate to **http://localhost:8088** (change **localhost** to the IP address, hostname, or domain name of your BI Platform).

2. Log in as **admin**, using **password** as the password.

3. Click the **+** (plus) button in the upper right corner of the Subscriptions section.

   An **Install License** dialogue will appear.

4. Click **Browse**, then navigate to the location you saved your LIC files to, click on one of the LIC files, 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.

5. Click **OK**.

   The Setup screen will change according to the LIC file you installed. **Repeat the LIC file installation process for each support entitlement you have purchased.**

   > **Note:** After installing license files, Linux and Solaris users must restart the application server before they can use any functions in the Pentaho User Console. The Pentaho Enterprise Console does not have to be restarted, so you can continue configuring the system, but you must restart JBoss or Tomcat when you are finished.

6. In the **Solution Directory** field, enter `/pentaho/server/biserver-ee/pentaho-solutions/`, or whatever the path to your pentaho-solutions directory is.

7. In the **Backup Directory** field, type in the location that you'd like to save Pentaho Enterprise Console backup data to.

This can be any local directory that your application server has permissions to write to. You cannot, however, use relative paths in this or any other configuration field in this screen -- **all paths must be absolute**.

8. In the **Pentaho Web-App Path** field, enter `/tomcat/webapps/pentaho/,` or whatever the path to the unpacked pentaho.war directory is for your application server.

9. In the **Platform Administrator User Name** field, type in **admin**, or if you already have Pentaho User Console accounts established, type in the account name that you will use to manage reports and schedules.

   This user account does not have to exist yet, and you can change this value later.

10. Click **OK**.

   The rest of the settings in this screen do not need to change right now.

11. Click **Configuration** in the menu on the left side of Pentaho Enterprise Console.

12. Click the **Web Settings** tab at the top of the screen.

13. Change the **Fully Qualified Server URL** setting to match your server's hostname, domain name, or IP address. Do not change the directory or port number.

   If you need to change the port number, consult the *Pentaho BI Suite Administrator's Guide*.

Your Pentaho BI Server now has a minimal configuration. If you need to return to this screen, click on the wrench/screwdriver icon in the upper right corner of the screen.

Your administration work is only beginning; you should now consult the *Pentaho BI Suite Administrator's Guide* to learn your way around the Pentaho Enterprise Console.

Starting the BI Server At Boot Time On OS X

You can start and stop the BI Server at any time by running the **start-pentaho** and **stop-pentaho** scripts. To start the Tomcat server automatically at boot time, and stop automatically during shutdown, follow the below procedure.

1. Navigate to the `/Library/StartupItems/` directory.

2. Create a new subdirectory called **Pentaho**.

3. Create a new file named **Pentaho** in the `/Library/StartupItems/Pentaho/` directory.

4. Copy the following text into the new Pentaho file, changing **/Applications/pentaho/server/biserver-ee** to the directory you untarred the BI Server package to:

```sh
#!/bin/sh
#
# /Library/StartupItems/Pentaho/Pentaho
#
# A script to automatically start up Pentaho on system bootup
# for Mac OS X.
#
# (c) 2010-2011 Pentaho

# Suppress the annoying "$1: unbound variable" error when no option
# was given
if [ -z $1 ] ; then
    echo "Usage: $0 [start|stop|restart] "
    exit 1
fi

# Source the common setup functions for startup scripts
test -r /etc/rc.common || exit 1
. /etc/rc.common

StartService ()
{
    ConsoleMessage "Starting Pentaho server"
    cd /Applications/pentaho/server/biserver-ee
    bash ./start-pentaho.sh
}

StopService ()
{
    ConsoleMessage "Stopping Pentaho server"
```

```
    cd /Applications/pentaho/server/biserver-ee
    bash ./stop-pentaho.sh
}

RestartService ()
{
    ConsoleMessage "Restarting Pentaho server"
    cd /Applications/pentaho/server/biserver-ee
    bash ./stop-pentaho.sh
    bash ./start-pentaho.sh
}

RunService "$1"
```

5. Create a new file called **StartupParameters.plist**, and copy the following text into it:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple Computer//DTD PLIST 1.0//EN" "http://www.apple.com/
DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
    <key>Description</key>
    <string>Pentaho 3.8</string>
    <key>OrderPreference</key>
    <string>None</string>
    <key>Provides</key>
    <array>
        <string>Pentaho</string>
    </array>
    <key>Uses</key>
    <array>
        <string>Network</string>
        <string>Resolver</string>
    </array>
</dict>
</plist>
```

6. Reboot the server to verify that the BI Server starts as intended.

   When the computer next starts, if you get a warning about the security settings of the new Pentaho startup item. choose the **Fix** option and then click **Restart**.

The Pentaho BI Server will now start at boot time.

You can check the status of the Pentaho server after by viewing **system.log** using the console viewer in Applications/ Utilities/Console.

Installing the Client Tools

You should already have downloaded the necessary client tool packages, so all you have to do now is untar them to the Applications folder on each of the appropriate desktop machines. There are no further required steps for installing the Pentaho client tools, but you may want to create desktop shortcuts or dock icons for them to make it easier for your users to launch them.

Testing and Using Your Server

Open a Web browser and navigate to the Pentaho BI Server Web address. If you installed the application onto your local machine, then the URL to reach the reporting and analysis samples is **http://localhost:8080/pentaho**. If you are trying to reach the application from a remote machine, the URL will be **http://example.com:8080/pentaho**, substituting **example.com** with the remote machine's hostname, IP address, or domain name. When the page loads, you should see a Web page similar to the below picture:

You won't be able to log into the Pentaho User Console until you've established user accounts and roles in the Pentaho Enterprise Console.

When your system is properly configured, log in with your administrator account, then verify that you can do the following, assuming your data has been prepared for reporting and analysis:

- Create a new analysis view.
- Create a new ad hoc report.
- If you are a Dashboards Enterprise Edition customer, create a new dashboard with the Dashboard Designer plugin.
- If you are a Pentaho Analysis Enterprise Edition customer, create a new analysis report with the Pentaho Analyzer plugin.
- Schedule a report to run at a regular interval.
- Using Pentaho Report Designer, publish a report to the BI Server.

Troubleshooting

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

Version Check

The instructions in this guide are specific to the Pentaho BI Server Enterprise Edition version 3.8.0-GA. The installation process can change significantly between BI Server releases to address new features, updated requirements, and bug workarounds, so the instructions in this guide should be assumed not to work with any other BI Server version, including the open source BI Server Community Edition version 3.8-stable.

Examining Log Files

If the BI Server fails to start or work properly, the log file you should consult is **pentaho.log** in the `/pentaho/server/biserver-ee/tomcat/bin/` directory. The contents of this file will assist you in tracking down the problem.

File Names and Paths

Note: This is the most common installation problem.

Many of the configuration files and paths in this guide are similar, and it is easy to confuse them, which could result in modifying the wrong files or copying to the wrong locations. Double-check your file names and paths and ensure that you've copied all of the right files to all of the correct directories.

Trailing slashes are important; both their inclusion and their absence, depending on the file and parameter or element you are modifying. Follow the examples in this guide exactly unless otherwise directed.

JDBC Driver Problems

First, ensure that the correct JDBC driver JARs are installed to the correct locations, then check to make sure that there aren't conflicting driver versions. The BI Server installation instructions explain how to add driver JARs to the correct locations; there is also a section titled **Adding a JDBC Driver** in the *Pentaho BI Suite Administrator's Guide* that explains driver locations for all parts of the BI Suite.

Note: Some database vendors (and driver developers) require using JDBC version 4 drivers with a Java 6 environment. Check with your database or driver vendor if you suspect you have having JDBC driver compatibility issues.

Cannot Create Hibernate Tables in MySQL

The Pentaho solution repository uses long text strings that require a longer maximum character limit than the default UTF-8 configuration allows. Therefore, if your MySQL character set is configured to use UTF-8, you must change it to **ASCII** instead in order to use it as a Pentaho solution database. Using UTF-8 will prevent the MySQL initialization scripts from running during installation.

Unable to Use the Database Init Scripts for PostgreSQL

The **pg_hba.conf** file contains host-based authentication information. If you can't run the SQL scripts that generate the Hibernate and Quartz databases, it's probably because the default user accounts for each database don't have the right permissions. To change this, edit the file to ensure that connections from local users created by the Pentaho sql scripts (**hibuser** and **pentaho_user**) will be able to connect. The default on Debian-based systems is for local connections yo use **ident** authentication, which means that database users must have local user accounts. In other words, to continue using **ident**, you would have to create local **hibuser** and **pentaho_user** accounts. It's easier to just change the authentication method to something less restrictive, if your IT manager permits you to do so.

context.xml Changes Do Not Take Effect After Re-deploying a WAR

Re-deployment of a WAR or EAR with a custom **context.xml** will, in some cases, cause the original context.xml that you deployed with the original WAR or EAR to become permanently cached. Tomcat in particular will generate a WAR-specific context configuration file, and keep it in place even after the WAR is deleted. The location and naming convention for this file are: **$CATALINA_HOME/conf/Catalina/<host>/<war name>.xml**. Typically this will be something like: **/tomcat/conf/Catalina/localhost/pentaho.xml**. If this file exists, you will have to delete it prior to re-deploying pentaho.war if you have made any changes to context.xml.