



Upgrading to the Pentaho BI Suite 4.1



This document is copyright © 2011 Pentaho Corporation. No part may be reprinted without written permission from Pentaho Corporation. All trademarks are the property of their respective owners.

Help and Support Resources

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

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

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

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

Limits of Liability and Disclaimer of Warranty

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

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

Trademarks

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

Company Information

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

E-mail: communityconnection@pentaho.com

Sales Inquiries: sales@pentaho.com

Documentation Suggestions: documentation@pentaho.com

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

Contents

Introduction.....	4
New Features in Version 4.1.....	4
Best Practices for Production Upgrades.....	5
Prerequisites.....	5
System Requirements.....	5
How to Check Your Java Version.....	5
Backing Up.....	6
Stopping the Solution Database (MySQL) Service on Windows.....	6
Renaming Your Old Pentaho Installation.....	7
Upgrading the BI Server.....	8
Manual WAR Deployment.....	8
Manual Deployment Upgrade Checklist.....	8
Downloading and Unpacking the Upgrade Materials.....	10
Removing the Xalan JAR for JBoss Deployments.....	10
Copying the Oracle Quartz JAR Pre-Build.....	10
Installing the Analysis Enterprise Edition Plugin.....	10
Using an Existing JBoss Instance.....	12
Using an Existing Tomcat Instance.....	13
Rebuilding the Solutions Directory.....	13
Installing Pentaho User Console Plugins.....	14
Installing Hadoop Hive Plugins.....	14
Archive-Based Deployment.....	15
Archive Deployment Upgrade Checklist.....	15
Obtaining the Installation Materials.....	16
Deploying a New BI Server.....	17
BI Suite Graphical Installer Deployment.....	22
Graphical Installer Deployment Upgrade Checklist.....	22
Obtaining the Installation Materials.....	24
Deploying the BI Server.....	25
Migrating Your Solutions.....	31
Merging Your BI Server Configuration Files.....	32
Updating pentahoObjects.spring.xml.....	32
Updating web.xml.....	33
Updating mondrian.properties.....	33
Upgrading Your Single-Sign On Configuration.....	34
CAS Property Reference.....	34
Upgrading the Pentaho Enterprise Console.....	36
Moving JDBC Drivers.....	37
Testing Your New BI Server.....	38
Upgrading the Pentaho Client Tools.....	39
Post-Upgrade Configuration.....	40
Cleanup.....	41
Troubleshooting.....	42
File Names and Paths.....	42
Examining Log Files.....	42
JDBC Driver Problems.....	42
Unable to Use the Database Init Scripts for PostgreSQL.....	42
context.xml Changes Do Not Take Effect After Re-deploying a WAR.....	42
Tomcat Logs Report Memory Leaks.....	43

Introduction



Danger: Do not skip ahead! Optional procedures are clearly defined. If you omit any required part of the upgrade process, you **will** be left with a non-functioning configuration and will very likely have to restore your previous configuration and start over from the beginning.

This guide provides procedures and advice for sysadmins who are upgrading from the Pentaho BI Suite Enterprise Edition 4.0.0-GA to version 4.1.0-GA. This encompasses the BI Server, DI Server and any Pentaho client tools that you may have purchased support modules for. **This guide is not intended for upgrades to or from any other BI Suite version, including release candidates (RC) and betas.**



Note: If you only need to update the Pentaho Data Integration server and client tools, you may find it easier to follow the *Pentaho Data Integration 4.2.0 to 4.2.1 Upgrade Guide* instead of trying to follow only the PDI sections of this larger BI Suite document.

You will upgrade to the latest BI and DI Server versions using either archive packages, or the manual WAR build and deployment process (BI Server only). Once the new version is installed, you'll migrate your solutions and server configuration settings from the old instance to the new one. Upgrading the Pentaho client tools is a much simpler operation that will take less time and effort. If you have a large number of client tool deployments, however, you may want to find a way to automate part or all of the client tool upgrade process.

New Features in Version 4.1

The Pentaho BI Suite 4.1 contains many new features and improvements:

- **Improved documentation:** Many documents have been updated to address customer feedback submitted through the Pentaho support and pre-sales engineering departments, and to accommodate the new features and capabilities introduced in this release. Most notably, there is extensive new documentation on Analysis (Mondrian) engine cache control. Several bugs have been fixed in the Pentaho documentation PDF style sheet.
 - **Impact:** All BI Suite users should be made aware of the location of Pentaho Documentation. If necessary, you may need to print or email and distribute a certain subset of PDFs to each user to ensure that they have the best self-help resources available to them.
- **Interactive Reporting:** Several important patches for bugs reported in BI Suite 4.0.
 - **Impact:** No user, developer, or administrator impact.
- **Data Integration 4.2.1:** Many patches for issues reported by Pentaho customers and community members. Upgraded to Mondrian 3.3 for related steps and Agile BI perspective. Added the ability to set the optimization level of the Modified Java Script Value step
 - **Impact:** BI developers who use the Modified Java Script Value step may need to examine their transformations to see if they can take advantage of the new optimization level setting.
- **Pentaho Analyzer improvements:** Support for segment cache log display from within Analyzer.
 - **Impact:** BI developers and administrators can now use Analyzer to view segment cache logs. Refer to the *Pentaho Analysis Guide* for more information.
- **BI Server 3.10:** Several important patches for bugs reported in BI Server 3.9.
 - **Impact:** No user or developer impact. Administrators performing BI Server upgrades should be aware that the process may differ if they have already applied Pentaho-supplied patches for BI Server 3.9.
- **Pentaho Analysis (Mondrian) Enterprise Edition:** Introduction of an official Analysis engine segment cache framework supporting Infinispan and Memcached, with the option of extending the engine to support other cache systems. Explain plan support has been added. Greenplum and Hive dialect support has also been added.
 - **Impact:** Pentaho Analysis Enterprise Edition customers now have expanded enterprise cache options that can dramatically increase Mondrian performance in large ROLAP deployments. Consult the *Pentaho Analysis Guide* to learn more about segment cache control and logging.

Best Practices for Production Upgrades

In order to minimize downtime for your users and ensure the integrity of your solution data, it's best to perform the upgrade on a test server that has an exact copy of your production BI Server. Once you've verified the upgrade process on your test machine, you can perform the same procedure on your production server with confidence.

You can work with your test server at any time without having to worry about disturbing your users, but you may want to schedule the production upgrade for a weekend, evening, or other period of planned downtime so that no users are interrupted in the process.

Some users may need additional training or documentation for Interactive Reporting, Pentaho Data Integration, and Analyzer. The latest Pentaho documentation can always be found in the Pentaho Knowledge Base at <http://kb.pentaho.com>. **In addition, Pentaho now provides most of its Enterprise Edition documentation in PDF format with every BI Server archive package and the graphical installer.** To access the PDFs, log into the Pentaho User Console and go to the **Help** menu. Most of the Pentaho client tools also ship with up-to-date PDF documentation.

To arrange for Web-based or classroom training sessions, contact your Pentaho sales representative.

Prerequisites

To complete the upgrade, you will need the following:


- A currently running Pentaho BI Server Enterprise Edition version 3.9.0-GA. **Other BI Server versions are not covered or supported in this guide.**
- A test server or virtual machine that contains a software environment identical to your production server.
- The knowledge and permissions to access and make changes to your production server.
- Administrator permissions in your existing Pentaho Enterprise Console.
- Sufficient time to complete each phase of the upgrade. The exact amount of time depends on the complexity of your security infrastructure, the size of your solution repository database, and the speed of your server hardware. Typical upgrades should take between one and two hours. If you have many custom changes in your Pentaho configuration files, you can expect to spend about two hours on the upgrade.

System Requirements

Hardware and software requirements have not changed for the BI Suite, so your existing production hardware will probably not need to be upgraded. Virtually any Linux distribution; Solaris 10; OS X; and Windows XP, Vista, Server 2008, or 7 will work, as long as it has a Sun Java Runtime Environment (JRE) or Java Development Kit (JDK -- this is a strict requirement only for manual deployment upgrades) version 1.6. Manual deployment upgrades will also need Apache Ant version 1.7 in order to build a new Pentaho WAR package.

How to Check Your Java Version

The Pentaho BI Suite requires a Java Runtime Environment (JRE) or Java Development Kit (JDK) version 1.6 (sometimes referred to as 6.0). Follow the procedure below to see which version of Java is installed on your system and configured to be the default Java executable.

 **Note:** There may be multiple JREs or JDKs on your system, but only one can be set as the global default. Any of these Java instances can be explicitly used for any Java program; if no specific Java executable is called, the default is used. Pentaho establishes a specific system variable named **PENTAHO_JAVA_HOME** to declare which Java instance it will use.

1. Open a terminal or command prompt window.
2. Type this command in: **java -version** and press **Enter**.

Along with the Java version, the bit-ness (32-bit or 64-bit) and patch level will also show in the output.

```
java version "1.6.0_21"  
Java(TM) SE Runtime Environment (build 1.6.0_21-b06)  
Java HotSpot(TM) 64-Bit Server VM (build 17.0-b16, mixed mode)
```

Backing Up



Note: You must quit all Pentaho programs and stop all Pentaho-related services (including the BI Server, DI Server, Pentaho Enterprise Console, and the solution database) before creating backups. After this point, do not start any of these programs or services until directed to do so.

Before you proceed with a production upgrade, you should back up every part of the BI Server that will change. The easiest way to back up a directory is to create a compressed archive of it. Generally on a Windows system you would use the Zip or RAR formats through programs like 7-zip or WinRAR. On Linux, BSD, and Solaris, you would use GNU Tar or bsdtar. No matter which program you use, you should end up with either one single file that encompasses the entire `/pentaho/` directory (for graphical installer and archive deployments), or the `/pentaho/` directory plus the **pentaho.war** file from your application server (for manual deployments).

Stopping the Solution Database (MySQL) Service on Windows

If you installed the BI Server on Windows through the graphical installation utility, the MySQL solution database will be running as a Windows service. To stop this service, go to the **Service Manager** in the Windows Control Panel, then stop the **Pentaho Solution Repository** service.

Renaming Your Old Pentaho Installation

Before you proceed, ensure that your BI Server, DI Server, and Pentaho Enterprise Console servers are stopped.

Rename your old `/pentaho/server/` directory to `/pentaho/server_old/` or some other appropriate name. Alternatively, you can move this directory to a different location.

```
mv /home/pentaho/pentaho/server/ /home/pentaho/pentaho/server_old/
```




Note: None of the software installed to this directory will work while it is renamed. If you need to abort the upgrade and start any servers in the server directory, you must rename it back to **server**.

Upgrading the BI Server

There are three upgrade paths for the BI Server, depending on how you installed the previous Pentaho release:


- 1. Manual deployment:** If you built a Pentaho WAR and deployed it to your own application server, you will follow the same process to upgrade it. After the WAR is deployed, you'll merge your old pentaho-solutions directory with the new one, then upgrade the Pentaho Enterprise Console. Client tools are upgraded separately by using archive packages. The instructions for this path begin here: [Manual WAR Deployment](#) on page 8.
- 2. Archive deployment:** If you deployed the BI Suite from individual archive packages, you'll download 4.1.0-GA packages, unpack them, and merge your old pentaho-solutions directory with the new one. The instructions for this path begin here: [Archive-Based Deployment](#) on page 15.
- 3. Graphical installer deployment:** If you previously used the graphical installation utility to install a BI Suite environment, the easiest upgrade path is to use a BI Server archive package to replace Enterprise Console and the BI Server, then merge your old pentaho-solutions directory with the new one. The instructions for this path begin here: [BI Suite Graphical Installer Deployment](#) on page 22.

Refer to the sections below that refer to your chosen upgrade path.

 **Note:** While the BI Suite is at version 4.1.0, the BI Server is at version 3.10.0, and Pentaho Data Integration is at version 4.2.1.

Manual WAR Deployment

If you previously used the manual deployment process to build a BI Server WAR, you should use the same process to upgrade to 4.1.0-GA (BI Server 3.10).

 **Note:** If you aren't making any custom changes to the WAR, and you aren't running any other applications on your application server, it may be easier to use the archive-based deployment method instead. The BI Server archive package includes a preconfigured Tomcat 6.0 server with the pentaho.war already deployed.

Manual Deployment Upgrade Checklist

The Upgrade Checklist is a concise list of instructions intended to show a high-level overview of the upgrade process. It also serves as a method of verifying that each task is performed in the correct order. You may find it useful to print the checklist out and physically mark each step in the Done column as you complete it. **The checklist is not the complete instruction set;** consult the verbose instructions throughout this guide for more details on each step.

Step	Procedure	Done
Step 1	Download the upgrade materials from the Enterprise Edition FTP site or Pentaho Knowledge Base.	
Step 2	Stop your application server, the Pentaho Enterprise Console server, and your solution database.	
Step 3	Back up your <code>/pentaho/</code> directory and the Pentaho Web application (pentaho.war or the uncompressed WAR directory in your application server's deploy or webapps directory).	
Step 4	Rename the <code>/pentaho/server/</code> directory to <code>/pentaho/server_old/</code> .	
Step 5	Unpack the biserver-manual-ee package to a temporary location.	
Step 6	Remove the BI Server's Xalan JAR from the build materials.	
Step 7	If you are using an Oracle solution repository, copy the quartz-oracle-1.5.2.jar file to the appropriate directory to be included in the WAR build.	
Step 8	If you have made any changes to web.xml , merge your customizations with the <code>/biserver-manual-ee/build-resources/pentaho-webapp/WEB-INF/web.xml</code> file, then copy that merged file to the <code>/biserver-manual-ee/build-resources/WEB-INF/</code> directory.	
Step 9	Copy your old context.xml file (for Tomcat deployments), or your pentahoHibernate-ds.xml and quartz-ds.xml files (for JBoss deployments) to the <code>/biserver-manual-ee/build-resources/custom-pentaho-webapp/META-INF/</code> directory.	
Step 10	If you have a Pentaho Analysis Enterprise Edition license, unpack the pentaho-analysis-ee archive to a temporary location, then copy the requisite JARs to the <code>/biserver-manual-ee/build-resources/custom-pentaho-webapp/WEB-INF/lib/</code>	

Step	Procedure	Done
	directory, and the requisite configuration files to the <code>/biserver-manual-ee/build-resources/custom-pentaho-webapp/WEB-INF/classes/</code> directory.	
Step 11	Build new Pentaho WAR packages with the BI Server 3.10 manual deployment materials.	
Step 12	Remove the old <code>pentaho.war</code> and <code>pentaho-style.war</code> files or directories from your application server's webapps or <code>deploy</code> directory.	
Step 13	Delete any old Pentaho-related temporary files or directories created by your application server.	
Step 14	Copy the new WARs to the <code>/server/webapps/</code> directory in Tomcat, or the <code>/server/default/deploy/</code> directory in JBoss.	
Step 15	Create a new <code>/pentaho/server/biserver-ee/</code> directory in the same location as the old one that you renamed earlier, and move the <code>/biserver-manual-ee/build/pentaho-solutions/</code> directory to it.	
Step 16	Unpack any Pentaho User Console plugins you may have licenses for (Analyzer, Dashboard Designer, Interactive Reporting) to the <code>/pentaho/server/biserver-ee/pentaho-solutions/system/</code> directory.	
Step 17	If you have a Pentaho BI Suite For Hadoop license, unpack the <code>pdi-hadoop-plugin</code> and <code>pdi-hadoop-plugin-ee</code> archives to the <code>/pentaho/server/biserver-ee/pentaho-solutions/system/kettle/plugins</code> directory.	
Step 18	Copy all of your solutions directories from the <code>/pentaho/server_old/biserver-ee/pentaho-solutions/</code> directory to the new one.	
Step 19	Copy the files documented in Merging Your BI Server Configuration Files on page 32 to the new <code>pentaho-solutions</code> directory.	
Step 20	Edit the following files and merge in the changes as instructed: <code>pentahoObjects.spring.xml</code> , <code>mondrian.properties</code> , and <code>web.xml</code> .	
Step 21	Perform the documented steps for upgrading single-sign on if you previously used this as your authentication method.	
Step 22	If you're a PDI Enterprise Edition customer, continue on to the section that describes the DI Server upgrade process.	
Step 23	Unpack the new Pentaho Enterprise Console from the archive package to the <code>/pentaho/server/</code> directory.	
Step 24	Copy the following files from the old Pentaho Enterprise Console to the new one: <code>console.xml</code> , <code>console.properties</code> , <code>login.properties</code> , <code>login.conf</code> , <code>log4j.xml</code> ; and the entire <code>/resource/config/hsqldb/</code> directory.	
Step 25	If you have not already done so, move data source JDBC driver JARs from the old directories to the new ones. This includes the BI Server and Enterprise Console.	
Step 26	If you installed the Interactive Reporting plugin, use the license installer utility (or start the BI Server and Enterprise Console and use the license interface in Enterprise Console) to install the Pentaho Reporting Enterprise Edition license key.	
Step 27	Start your solution database, Tomcat or JBoss service, and the Pentaho Enterprise Console and check for any obvious errors or failures in the console output or log file.	
Step 28	Ensure that all of your content is accessible. Log into the Pentaho User Console and verify that your product licenses work. Run a new report in Interactive Reporting to ensure that it is ready for production. Verify at least two existing analysis views or Analyzer reports for each data source. Verify that any previously created dashboards are functioning properly. Run all reports that have been published from Report Designer. Lastly, check all of your schedules to ensure that they are still properly configured and operational.	
Step 29	On workstations, unpack applicable client tools to the empty <code>/pentaho/design-tools/</code> directory.	
Step 30	Verify that all client tools can access old content and publish to the BI Server. Verify that PDI can connect to the DI Server, if applicable.	
Step 31	Remove the <code>/pentaho/server_old/</code> directory on each upgraded machine, and any archive packages and temporary directories you used during the upgrade.	

Downloading and Unpacking the Upgrade Materials

This procedure covers BI Server manual deployment only. You can update the Pentaho client tools on your user workstations with the graphical installer, or you can download each client tool as a zip or tar.gz archive and deploy them individually.

Log into the Pentaho Enterprise Edition FTP site to download the upgrade packages. The address and login credentials were emailed to you with your new Welcome Kit. If you cannot find or were not sent a Welcome Kit, contact your Pentaho sales or support representative.

These are the files you need:

1. /4.1.0-GA/developers/biserver-manual-ee-3.10.0-GA.zip
2. /4.1.0-GA/server/windows/pec-4.1.0-GA.zip or /4.1.0-GA/server/nix_osx/pec-4.1.0-GA.tar.gz for Linux, OS X and Solaris.
3. If you purchased Dashboards, Reporting, or Analysis support entitlements, you will also need the appropriate plugins in /4.1.0-GA/server/plugins/
4. Optionally, for client tool deployments, navigate to /4.1.0-GA/installers/ and download the BI Suite installer for your platform -- Windows, Linux, or OS X.

These files contain the BI Server manual deployment package; the Pentaho Enterprise Console; the Dashboard Designer, Interactive Reporting, and Pentaho Analyzer plugins; and the BI Suite installer, respectively. The BI Server and Pentaho Enterprise Console packages are required; Dashboard Designer, Interactive Reporting, and Analyzer are only for customers who have a support entitlement for them; and the BI Suite graphical installer is for client tool deployments and is not required for a server upgrade. If you plan to unpack individual client tool packages on your workstations, download those instead of the graphical installer.

Once the files are downloaded, unpack the biserver-manual-ee file first. It doesn't matter where you unpack it to -- it is only a temporary directory that you will use to build a new Pentaho Web application.

Next, if you downloaded any BI Server plugins, unpack them into the newly created /biserver-manual-ee/build-resources/pentaho-solutions/system/ directory.

You'll unpack the Pentaho Enterprise Console archive later when you're ready to replace the old one. For now, just leave it alone.


Removing the Xalan JAR for JBoss Deployments

JBoss 5.1 includes a Xalan JAR file that interferes with one included in the BI Server; they are different versions, so there is a class-cast exception whenever Xalan is called by BI Server content.

To fix this problem, remove the Xalan JAR from the BI Server build materials before building the WAR:

```
rm ./biserver-manual-ee/build-resources/pentaho-third-party/xalan-2.4.1.jar
```

This will force the BI Server to use the newer JBoss-supplied Xalan instance.

 **Note:** If you've already built a WAR and don't want to rebuild it, you can unpack the WAR and remove this JAR from the lib directory by hand.

Copying the Oracle Quartz JAR Pre-Build

This section is only for administrators who are using an Oracle database for the Pentaho solution repository. You do not need to do this for Oracle data sources; only for Oracle solution repositories.

1. Create a /biserver-manual-ee/build-resources/custom-pentaho-webapp/WEB-INF/lib/ directory.
2. Copy the /biserver-manual-ee/build-resources/pentaho-third-party/quartz-oracle-1.5.2.jar to the newly created /biserver-manual-ee/build-resources/custom-pentaho-webapp/WEB-INF/lib/ directory.

You are now ready to proceed to the WAR build instructions.

Installing the Analysis Enterprise Edition Plugin

Follow the instructions below to install the Pentaho Analysis Enterprise Edition package, which contains Analysis engine enhancements for large ROLAP deployments.



Note: This procedure does not cover Pentaho Analyzer installation.



Note: If you are performing a manual WAR build and deployment, and you want to slipstream the Pentaho Analysis Enterprise Edition JARs into the WAR pre-build, you can substitute `/biserver-manual-ee/build-resources/custom-pentaho-webapp/WEB-INF/` for `/WEB-INF/` references below.

1. If you have not already done so, retrieve the **pentaho-analysis-ee-3.3.0-GA** package from the Pentaho Enterprise Edition FTP site (login credentials were emailed to you with your Welcome Kit).
2. Unpack the archive to a temporary location.
3. If it is currently running, shut down the BI Server.
4. Copy only the following JARs from the `/pentaho-analysis-ee/lib/` directory to the `/WEB-INF/lib/` directory inside of the deployed `pentaho.war`:
 - `infinispan-core-4.2.1.FINAL`
 - `jboss-transaction-api-1.0.1.GA`
 - `jcip-annotations-1.0`
 - `jgroups-2.12.0.CR5`
 - `marshalling-api-1.2.3.GA`
 - `memcached-0.0.1-PENTAHO`
 - `pentaho-analysis-ee-3.3.0-GA-obf`
 - `rhq-pluginAnnotations-3.0.1`
 - `river-1.2.3.GA`
5. Copy all of the configuration files from `/pentaho-analysis-ee/config/` to the `/WEB-INF/classes/` directory inside of your deployed `pentaho.war`.
6. Remove the temporary **pentaho-analysis-ee** directory.

Pentaho Analysis Enterprise Edition is now installed with the default Infinispan configuration.

If you need to switch from Infinispan to Memcached, continue to the next section.

Switching to Memcached

In order to complete this procedure, you must have your own pre-configured Memcached instance. You should have also installed the Analysis Enterprise Edition package to your BI Server or standalone Mondrian engine.

If you already use the Memcached cache framework in your organization and would like to hook it up to the Pentaho Analysis ROLAP engine, follow the directions below to switch from the default Infinispan cache framework.

1. If the BI Server or standalone Mondrian engine are running, shut them down now.
2. If you performed a default install of the Pentaho Analysis Enterprise Edition package, then you should have all of the required JARs installed to the BI or Mondrian server. If you aren't sure, verify now that the following JARs are present in the `/WEB-INF/lib/` directory inside of your deployed `pentaho.war` or Mondrian engine:
 - `pentaho-analysis-ee`
 - `commons-lang`
 - `commons-io`
 - `commons-codec`
 - `pentaho-ee-dsc-core`
 - `memcached`
3. Edit the **pentaho-analysis-config.xml** in the `/WEB-INF/classes/` directory inside the deployed `pentaho.war` or Mondrian engine, and change the value of **SEGMENT_CACHE_IMPL** to match the class name referenced below:

```
<entry key="SEGMENT_CACHE_IMPL">com.pentaho.analysis.segmentcache.impl.memcached.MemcachedSegmentCache</entry>
```

4. Edit the **memcached-config.xml** in the `/WEB-INF/classes/` directory inside the deployed `pentaho.war` or Mondrian engine, and change the values of **SALT**, **SERVERS**, and **WEIGHT** to match your preference:

Refer to [Memcached Configuration Options](#) on page 12 for more information on these settings.


```
<entry key="SALT">YOUR SECRET SALT VALUE HERE</entry>
```

```
<entry key="SERVERS">192.168.0.1:1642,192.168.0.2:1642</entry>
```

```
<entry key="WEIGHTS">1,1</entry>
```

Your Pentaho Analysis Enterprise Edition instance is now configured to use Memcached for ROLAP segment caching. **Memcached Configuration Options**


These properties control Memcached settings, and are set in the **memcached-config.xml** file in the `/WEB-INF/classes/` directory inside of your deployed pentaho.war or Mondrian engine.

 **Note:** This is not a comprehensive list of the potential Memcached settings; the options explained below are the ones most critical to Memcached configuration for Pentaho Analysis. You must research further Memcached options on your own.

Property	Purpose
SERVERS	A comma-separated list of servers and port numbers representing the Memcached nodes usable by the plugin.
WEIGHTS	A comma-separated list of numbers representing the relative caching capacity of the servers defined in the SERVERS property. There must be exactly as many values of WEIGHTS as there are values of SERVERS. As an example, if the first server has a capacity of 128 megabytes, and the second has a capacity of 256 megabytes, the correct values for the WEIGHTS property should be "1,2", indicating that the first server has a relative size of half of the second one.
SALT	A secret key prefix to be used when saving and loading segment data from the Memcached nodes. This property must be the same for all Mondrian nodes that share their caches. If the SALT value is different from one node to the next, the nodes will not be able to share their cache data.

Using an Existing JBoss Instance

Upgrade instructions for JBoss WARs are below.

 **Note:** Support for JBoss versions prior to 5.1 is deprecated. You must use JBoss 5.1 for this upgrade.

Building and Deploying a New JBoss WAR

Ensure that the application server and solution database are halted before continuing with the upgrade.

This process walks you through building a new pentaho.war and deploying it to your application server. You will have to adjust the paths in the examples to match your environment.

1. If you have made any customizations to your old **web.xml** file in the `/jboss/server/default/deploy/pentaho.war/WEB-INF/` directory, merge them into the new **web.xml** in `/biserver-manual-ee/build-resources/pentaho-webapp/WEB-INF/` and copy the merged file to `/biserver-manual-ee/build-resources/custom-pentaho-webapp/WEB-INF/`.
2. Open a terminal window and navigate to the `/biserver-manual-ee/` directory.
3. Run the following Ant command, changing the database to whichever one you're currently using for your solution repository: `ant -Darchive.target=war-pentaho-jboss-no-portal -Ddb=mysql5`

This builds an application server- and database-specific WAR. Possible database values are:


- mysql5
 - postgresql
 - oracle10g
4. Delete all of the old Pentaho WARs from the `/jboss/server/default/deploy/` directory. The relevant files are: `pentaho.war` and `pentaho-style.war`.
 5. Copy the newly-built `pentaho.war` file from the `/biserver-manual-ee/build/pentaho-wars/jboss/portal/` directory to your existing `/jboss/server/default/deploy/` directory.

6. Copy the newly-built `pentaho-style.war` file from the `/biserver-manual-ee/build/pentaho-wars/` directory to your existing `/jboss/server/default/deploy/` directory.
7. Delete the `/jboss/server/default/tmp/` and `/jboss/server/default/work/` directories.
These directories contain temporary files that, when missing, will force your application server to recompile all of your JSPs. This is a necessary step and will not cause any harm to JBoss.

Your instance of the Pentaho BI Server is now updated to version 4.1, though there is much more to do before your upgrade is complete.

Using an Existing Tomcat Instance

Upgrade instructions for Tomcat WARs are below.


 **Note:** Support for Tomcat 5.5 is deprecated. You must use Tomcat 6 for this upgrade.

Building and Deploying a New Tomcat WAR

Ensure that the application server and solution database are halted before continuing with the upgrade.

This process walks you through building and deploying a new BI Server WAR archive for Tomcat 6. You will have to adjust the paths in the examples to match your environment.

1. Copy your current `context.xml` file from your old Pentaho WAR directory to the **META-INF** subdirectory in `/biserver-manual-ee/build-resources/custom-pentaho-webapp/`.

 **Note:** If you have made any changes to `context.xml`, you must delete the `pentaho.xml` file that Tomcat automatically generates based on the `context.xml` embedded in the Pentaho WAR. Tomcat will not overwrite this file on its own, so any `context.xml` changes will not be recognized until the old `pentaho.xml` is removed. Typically this file is located in `/tomcat/conf/Catalina/localhost/`. If this file is not present, then you probably didn't make any changes to `context.xml` and there is nothing to worry about.

```
cp /pentaho/server/biserver-ee/tomcat/webapps/pentaho/META-INF/context.xml /home/pgibbons/biserver-manual-ee/build-resources/custom-pentaho-webapp/META-INF/
```

2. If you have made any customizations to your old `web.xml` file in the `/tomcat/webapps/pentaho/WEB-INF/` directory, merge them into the new `web.xml` in `/biserver-manual-ee/build-resources/pentaho-webapp/WEB-INF/` and copy the merged file to `/biserver-manual-ee/build-resources/custom-pentaho-webapp/WEB-INF/`.
3. Delete the unpacked WAR directories in `/tomcat/webapps/`, which are: **pentaho** and **pentaho-style**.
4. Open a terminal window and navigate to the `/biserver-manual-ee/` directory.
5. Run the following Ant command, changing the database to whichever one you're currently using for your solution repository: `ant -Darchive.target=war-pentaho-tomcat -Ddb=mysql5`

This builds an application server- and database-specific WAR. Supported database values are:

- mysql5
- postgresql
- oracle10g

6. Copy the newly-built `pentaho.war` from the `/biserver-manual-ee/build/pentaho-wars/tomcat/` directory to your existing `/tomcat/webapps/` directory.
All of the WARs you deploy will automatically unpack when Tomcat starts, and create new directories to replace the ones you deleted previously.
7. Copy the `pentaho-style.war` file from the `/biserver-manual-ee/build/pentaho-wars/` directory to your existing `/tomcat/webapps/` directory.

Your instance of the Pentaho BI Server is now updated to version 3.10, though there is much more to do before your upgrade is complete.

Rebuilding the Solutions Directory

Follow the directions below to create a new `pentaho-solutions` directory. Modify the paths in the examples to match your scenario.

1. Create a new `/pentaho/server/biserver-ee/` directory in the exact location of the old one, which you earlier renamed to `/pentaho/server_old/biserver-ee/`.

```
mkdir -p /home/pentaho/pentaho/server/biserver-ee/
```

2. Copy the newly-built pentaho-solutions directory from `/biserver-manual-ee/build/` to the `/pentaho/server/` directory you just created.

```
cp -r /home/amenethil/downloads/biserver-manual-ee/build/pentaho-solutions/ /home/pentaho/pentaho/server/biserver-ee/
```


You now have a fresh solutions directory. You can now install the latest plugins, and migrate your settings and solutions to it according to the instructions later in this guide.

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
- Pentaho Interactive Reporting

 **Note:** The Pentaho Analysis Enterprise Edition plugin is separate from Analyzer, and requires a different installation process explained in another part of this guide.

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**; the Interactive Reporting plugin always starts with **pir**.

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

This will create a subdirectory for each plugin that you unpack.

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. If this directory does not exist, create it.

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.

Archive-Based Deployment

To upgrade the BI Server via archive packages, you'll simply download new packages, unpack them, and merge your BI Server configuration and solutions by hand. Step-by-step instructions are below.

Archive Deployment Upgrade Checklist

The Upgrade Checklist is a concise list of instructions intended to show a high-level overview of the upgrade process. It also serves as a method of verifying that each task is performed in the correct order. You may find it useful to print the checklist out and physically mark each step in the Done column as you complete it. **The checklist is not the complete instruction set**; consult the verbose instructions throughout this guide for more details on each step.

Step	Procedure	Done
Step 1	Download the upgrade materials from the Enterprise Edition FTP site or Pentaho Knowledge Base.	
Step 2	Stop all Pentaho programs and services, including the solution repository database.	
Step 3	Back up your /pentaho/ directory and hibernate and quartz databases.	
Step 4	Rename the /pentaho/server/ directory to /pentaho/server_old/, then create a new, empty /pentaho/server/ directory.	
Step 5	Unpack the BI Server archive package to the empty /pentaho/server/ directory.	
Step 6	Copy any necessary JDBC drivers for your solution database and data sources to the /pentaho/server/biserver-ee/tomcat/lib/ directory.	
Step 7	Copy over your old web.xml to the new /tomcat/webapps/pentaho/WEB-INF/ directory.	
Step 8	Copy your old context.xml file from the /server_old/biserver-ee/tomcat/webapps/pentaho/META-INF/ directory to the new one.	
Step 9	Copy over your old server.xml to the new /tomcat/conf/ directory.	
Step 10	Unpack any Pentaho User Console plugins you may have licenses for (Analyzer, Dashboard Designer, Interactive Reporting) to the /pentaho/server/biserver-ee/pentaho-solutions/system/ directory.	
Step 11	If you have a Pentaho BI Suite For Hadoop license, unpack the pdi-hadoop-plugin and pdi-hadoop-plugin-ee archives to the /pentaho/server/biserver-ee/pentaho-solutions/system/kettle/plugins directory.	
Step 12	If you have a Pentaho Analysis Enterprise Edition license, unpack the pentaho-analysis-ee archive to a temporary location, then copy the requisite JARs to the /pentaho/server/biserver-ee/tomcat/webapps/pentaho/WEB-INF/lib/ directory, and the requisite configuration files to the /pentaho/server/biserver-ee/tomcat/webapps/pentaho/WEB-INF/classes/ directory.	
Step 13	Remove the Pentaho sample content.	
Step 14	Copy all of your solution directories from the /pentaho/server_old/biserver-ee/pentaho-solutions/ directory to the new one.	
Step 15	Copy the files documented in Merging Your BI Server Configuration Files on page 32 to the new pentaho-solutions directory.	
Step 16	Edit the following files and merge in the changes as instructed: pentahoObjects.spring.xml, mondrian.properties, and web.xml.	
Step 17	If you're a PDI Enterprise Edition customer, continue on to the section that describes the DI Server upgrade process.	
Step 18	Copy the following files from the old Pentaho Enterprise Console to the new one: console.xml, console.properties, login.properties, login.conf, log4j.xml; and the entire /resource/config/hsqldb/ directory.	
Step 19	If you have not already done so, move data source JDBC driver JARs from the old directories to the new ones. This includes the BI Server and Enterprise Console.	
Step 20	If you installed the Interactive Reporting plugin, use the license installer utility (or start the BI Server and Enterprise Console and use the license interface in Enterprise Console) to install the Pentaho Reporting Enterprise Edition license key.	
Step 21	Start the solution database, BI Server, and Pentaho Enterprise Console and check for any obvious errors or failures in the console output and log file.	

Step	Procedure	Done
Step 22	Check that all of your content is accessible. Log into the Pentaho User Console and verify that your product licenses work. Run a new report in Interactive Reporting to ensure that it is ready for production. Verify at least two existing analysis views or Analyzer reports for each data source. Verify that any previously created dashboards are functioning properly. Run all reports that have been published from Report Designer. Lastly, check all of your schedules to ensure that they are still properly configured and operational.	
Step 23	On workstations, remove old client tool directories, then unpack applicable client tools to the empty <code>/pentaho/design-tools/</code> directory.	
Step 24	Verify that all client tools can access old content and publish to the BI Server. Verify that PDI can connect to the DI Server, if applicable.	
Step 25	Remove the <code>/pentaho/server_old/</code> directory on each upgraded machine, and any archive packages and temporary directories you used for the upgrade.	

Obtaining the Installation Materials

As an Enterprise Edition customer, you can obtain the BI Server and client tool pre-configured packages from the Pentaho Enterprise Edition FTP site by using your 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.

With the exception of Design Studio, which has separate packages for Linux, OS X, and Windows, all of the below-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 admins. Other than this archive technology difference, the BI Suite packages are identical.

The Pentaho Enterprise Console is included in the BI Server package, so you do not need to download it separately.

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

Windows

The packages you will need for a Windows server are:

- **BI Server (includes the Pentaho Enterprise Console):** `/4.1.0-GA/server/windows/biserver-ee-3.10.0-GA.zip`
- **Pentaho Analyzer plugin:** `/4.1.0-GA/server/plugins/paz/`
- **Pentaho Dashboard Designer plugin:** `/4.1.0-GA/server/plugins/pdd/`
- **Pentaho Interactive Reporting plugin:** `/4.1.0-GA/server/plugins/pir/`
- **BI Server Hadoop plugin:** `/Pentaho_Data_Integration/4.2.1-GA/server/plugins/pdi-hadoop-plugin-1.0.3.zip`
- **BI Server Hadoop Enterprise Edition plugin:** `/Pentaho_Data_Integration/4.2.1-GA/server/plugins/pdi-hadoop-plugin-ee-1.0.4.zip`

The packages you will need for your Windows workstations are:

- **Report Designer:** `/4.1.0-GA/client/windows/prd-ee-4.1.0-GA.zip`
- **Metadata Editor:** `/4.1.0-GA/client/windows/pme-ee-4.1.0-GA.zip`
- ***Schema Workbench:** `/4.1.0-GA/client/windows/psw-ee-3.3.0.14682.zip`
- **Design Studio for 32-bit Windows:** `/4.1.0-GA/client/windows/pds-ee-win-32-4.0.0-GA.zip`
- **Design Studio for 64-bit Windows:** `/4.1.0-GA/client/windows/pds-ee-win-64-4.0.0-GA.zip`
- **Design Studio plugins for existing Eclipse installations:** `/4.1.0-GA/client/design-studio/org.pentaho.designstudio.editors.actionsequence_4.0.0.GA.zip`
- **Aggregation Designer:** `/4.1.0-GA/client/windows/pad-ee-1.3.0-GA.zip`
- **Data Integration:** `/4.1.0-GA/client/windows/pdi-ee-client-4.2.1-GA.zip`

Linux and Solaris


The packages you will need for a Linux or Solaris server are:

- **BI Server (includes the Pentaho Enterprise Console):** `/4.1.0-GA/server/nix/biserver-ee-3.10.0-GA.tar.gz`
- **Pentaho Analyzer plugin:** `/4.1.0-GA/server/plugins/paz/`
- **Pentaho Dashboard Designer plugin:** `/4.1.0-GA/server/plugins/pdd/`

- **Pentaho Interactive Reporting plugin:** /4.1.0-GA/server/plugins/pir/
- **BI Server Hadoop plugin:** /Pentaho_Data_Integration/4.2.1-GA/server/plugins/pdi-hadoop-plugin-1.0.3.zip
- **BI Server Hadoop Enterprise Edition plugin:** /Pentaho_Data_Integration/4.2.1-GA/server/plugins/pdi-hadoop-plugin-ee-1.0.4.zip


The packages you will need for your Linux or Solaris workstations are:

- **Report Designer:** /4.1.0-GA/client/nix_osx/prd-ee-4.1.0-GA.tar.gz
- **Metadata Editor:** /4.1.0-GA/client/nix_osx/pme-ee-4.1.0-GA.tar.gz
- ***Schema Workbench:** /4.1.0-GA/client/nix_osx/psw-ee-3.3.0.14682.tar.gz
- **Design Studio for 32-bit Linux:** /4.1.0-GA/client/nix_osx/pds-ee-linux-32-4.0.0-GA.tar.gz
- **Design Studio for 64-bit Linux:** /4.1.0-GA/client/nix_osx/pds-ee-linux-64-4.0.0-GA.tar.gz
- **Design Studio plugins for existing Eclipse installations:** /4.1.0-GA/client/design-studio/org.pentaho.designstudio.editors.actionsequence_4.0.0.GA.zip
- **Aggregation Designer:** /4.1.0-GA/client/nix_osx/pad-ee-1.3.0-GA.tar.gz
- **Data Integration:** /4.1.0-GA/client/nix_osx/pdi-ee-client-4.2.1-GA.tar.gz

 **Note:** The PSW package name's exact version number may change between the publication of this list and the GA release.

Deploying the BI Server

Follow the directions below to install a new BI Server and the Pentaho Enterprise Console. The examples assume a Linux or Solaris environment; adjust the paths accordingly for Windows-based machines.

 **Note: Solaris users:** The default behavior of the tar utility in Solaris 10 is to truncate long file names when unpacking an archive created with GNU tar, as the Pentaho archives are. Therefore, you must use GNU tar instead of Solaris tar for this procedure. The path to GNU tar on Solaris 10 is typically `/usr/sfw/bin/gtar`. If you do not have `gtar` installed on your system, you will need to install it or some other GNU tar-compatible utility (such as `star`).

1. Re-create the `/pentaho/server/` directory that you renamed earlier.

```
mkdir -p /home/pentaho/pentaho/server/
```

2. Untar or unzip the **biserver-ee-3.10.0-GA** archive from wherever you downloaded it to the new location.

```
tar zxvf /home/amenethil/downloads/biserver-ee-3.10.0-GA.tar.gz -C /home/pentaho/pentaho/server/
```

This will create new **biserver-ee** and **enterprise-console** subdirectories.

The BI Server and Pentaho Enterprise Console are now extracted into the proper directories. Continue on to the next several sections to install drivers, change the hostname and port if necessary, and install any plugins that you have support entitlements for.

Copying Solution Database JDBC Drivers

This is only for Pentaho solution databases. You will also need to install JDBC drivers for your data sources at a later time, but the instructions in this section are focused solely on solution database configuration and connectivity with the BI Server.

Follow the below process to enable the BI Server to connect to an Oracle, MySQL, or PostgreSQL solution database.

1. Find or retrieve a JDBC driver JAR from your database vendor or third-party driver developer.

Due to licensing restrictions, Pentaho is unable to provide the necessary JDBC driver JARs. You can retrieve a JDBC driver from your database vendor. To that end, you may find these links helpful:

- **Oracle:** http://www.oracle.com/technology/tech/java/sqlj_jdbc/index.html
- **MySQL:** <http://www.mysql.com/downloads/connector/j/>
- **PostgreSQL:** <http://jdbc.postgresql.org/download.html>

2. For the BI Server, copy the appropriate JDBC driver JAR file to the `/tomcat/lib/` directory for Tomcat, or the `/jboss/server/default/lib/` directory for JBoss.
3. For Enterprise Console, copy the driver JAR to the `/pentaho/server/enterprise-console/lib/` directory.

The BI Server and Enterprise Console now have the necessary driver to communicate with your solution repository database.

Copying context.xml, server.xml, and web.xml

The context.xml, server.xml, and web.xml files contain important configuration information that must be transferred to your new BI Server instance. Some of these files may need to be changed later in the upgrade process, but for right now, you must copy them to the new deployment.

1. Copy your old **context.xml** from `/server_old/biserver-ee/tomcat/webapps/pentaho/META-INF/` to `/server/biserver-ee/tomcat/webapps/pentaho/META-INF/`.
2. Copy your old **web.xml** from `/server_old/biserver-ee/tomcat/webapps/pentaho/WEB-INF/` to `/server/biserver-ee/tomcat/webapps/pentaho/WEB-INF/`.
3. Copy your old **server.xml** from `/server_old/biserver-ee/tomcat/conf/` to `/server/biserver-ee/tomcat/conf/`.


Your core BI Server configuration files have been transferred to your new BI Server.

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
- Pentaho Interactive Reporting

 **Note:** The Pentaho Analysis Enterprise Edition plugin is separate from Analyzer, and requires a different installation process explained in another part of this guide.

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**; the Interactive Reporting plugin always starts with **pir**.

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

This will create a subdirectory for each plugin that you unpack.

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. If this directory does not exist, create it.

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.

Installing the Analysis Enterprise Edition Plugin

Follow the instructions below to install the Pentaho Analysis Enterprise Edition package, which contains Analysis engine enhancements for large ROLAP deployments.

 **Note:** This procedure does not cover Pentaho Analyzer installation.

 **Note:** If you are performing a manual WAR build and deployment, and you want to slipstream the Pentaho Analysis Enterprise Edition JARs into the WAR pre-build, you can substitute `/biserver-manual-ee/build-resources/custom-pentaho-webapp/WEB-INF/` for `/WEB-INF/` references below.

1. If you have not already done so, retrieve the **pentaho-analysis-ee-3.3.0-GA** package from the Pentaho Enterprise Edition FTP site (login credentials were emailed to you with your Welcome Kit).
2. Unpack the archive to a temporary location.
3. If it is currently running, shut down the BI Server.
4. Copy only the following JARs from the `/pentaho-analysis-ee/lib/` directory to the `/WEB-INF/lib/` directory inside of the deployed `pentaho.war`:
 - `infinispan-core-4.2.1.FINAL`
 - `jboss-transaction-api-1.0.1.GA`
 - `jcip-annotations-1.0`
 - `jgroups-2.12.0.CR5`
 - `marshalling-api-1.2.3.GA`
 - `memcached-0.0.1-PENTAHO`
 - `pentaho-analysis-ee-3.3.0-GA-obf`
 - `rhq-pluginAnnotations-3.0.1`
 - `river-1.2.3.GA`
5. Copy all of the configuration files from `/pentaho-analysis-ee/config/` to the `/WEB-INF/classes/` directory inside of your deployed `pentaho.war`.
6. Remove the temporary **pentaho-analysis-ee** directory.

Pentaho Analysis Enterprise Edition is now installed with the default Infinispan configuration.

If you need to switch from Infinispan to Memcached, continue to the next section.

Switching to Memcached

In order to complete this procedure, you must have your own pre-configured Memcached instance. You should have also installed the Analysis Enterprise Edition package to your BI Server or standalone Mondrian engine.

If you already use the Memcached cache framework in your organization and would like to hook it up to the Pentaho Analysis ROLAP engine, follow the directions below to switch from the default Infinispan cache framework.

1. If the BI Server or standalone Mondrian engine are running, shut them down now.
2. If you performed a default install of the Pentaho Analysis Enterprise Edition package, then you should have all of the required JARs installed to the BI or Mondrian server. If you aren't sure, verify now that the following JARs are present in the `/WEB-INF/lib/` directory inside of your deployed `pentaho.war` or Mondrian engine:

- `pentaho-analysis-ee`
- `commons-lang`
- `commons-io`
- `commons-codec`
- `pentaho-ee-dsc-core`
- `memcached`

3. Edit the **pentaho-analysis-config.xml** in the `/WEB-INF/classes/` directory inside the deployed `pentaho.war` or Mondrian engine, and change the value of **SEGMENT_CACHE_IMPL** to match the class name referenced below:

```
<entry key="SEGMENT_CACHE_IMPL">com.pentaho.analysis.segmentcache.impl.memcached.MemcachedSegmentCache</entry>
```

4. Edit the **memcached-config.xml** in the `/WEB-INF/classes/` directory inside the deployed `pentaho.war` or Mondrian engine, and change the values of **SALT**, **SERVERS**, and **WEIGHT** to match your preference:
Refer to [Memcached Configuration Options](#) on page 12 for more information on these settings.


```
<entry key="SALT">YOUR SECRET SALT VALUE HERE</entry>

<entry key="SERVERS">192.168.0.1:1642,192.168.0.2:1642</entry>

<entry key="WEIGHTS">1,1</entry>
```

Your Pentaho Analysis Enterprise Edition instance is now configured to use Memcached for ROLAP segment caching.
[Memcached Configuration Options](#)

These properties control Memcached settings, and are set in the **memcached-config.xml** file in the `/WEB-INF/classes/` directory inside of your deployed `pentaho.war` or Mondrian engine.

 **Note:** This is not a comprehensive list of the potential Memcached settings; the options explained below are the ones most critical to Memcached configuration for Pentaho Analysis. You must research further Memcached options on your own.

Property	Purpose
SERVERS	A comma-separated list of servers and port numbers representing the Memcached nodes usable by the plugin.
WEIGHTS	A comma-separated list of numbers representing the relative caching capacity of the servers defined in the SERVERS property. There must be exactly as many values of WEIGHTS as there are values of SERVERS. As an example, if the first server has a capacity of 128 megabytes, and the second has a capacity of 256 megabytes, the correct values for the WEIGHTS property should be "1,2", indicating that the first server has a relative size of half of the second one.
SALT	A secret key prefix to be used when saving and loading segment data from the Memcached nodes. This property must be the same for all Mondrian nodes that share their caches. If the SALT value is different from one node to the next, the nodes will not be able to share their cache data.

Removing Pentaho Sample Data and Solutions

By default, Pentaho provides a sample data source and a solution directory filled with example content. These samples are provided for evaluation and testing. Once you are ready to move from an evaluation or testing scenario to development or production, you should remove the samples so that they do not get in the way of your BI content. Follow the instructions below to completely remove the Pentaho sample data and solutions.

1. Stop the BI Server.
2. If they exist, delete the `/pentaho/server/biserver-ee/pentaho-solutions/steel-wheels/` and `/pentaho/server/biserver-ee/pentaho-solutions/bi-developers/` directories.
These directories contain sample content for CDF and BI Server plugins. They are not present in typical manual deployment scenarios.
3. Edit the `/pentaho/WEB-INF/web.xml` file inside of the deployed `pentaho.war`.
As laid down by the Pentaho graphical installer and archive packages, this path should be `/pentaho/server/biserver-ee/tomcat/webapps/pentaho/WEB-INF/web.xml`. If you performed a manual WAR build and deployment, then you must adjust the path to fit your configuration.
4. Remove the sample servlets by commenting them out or removing them from the file:

```
<!-- [BEGIN SAMPLE SERVLETS] -->
<servlet>
  <servlet-name>UpdateSampleEmails</servlet-name>
  <jsp-file>/jsp/UpdateSampleEmails.jsp</jsp-file>
</servlet>

<servlet>
  <servlet-name>Widgets</servlet-name>
```

```

    <jsp-file>/jsp/Widgets.jsp</jsp-file>
</servlet>

<servlet>
  <servlet-name>SampleDashboard</servlet-name>
  <jsp-file>/jsp/SampleDashboard.jsp</jsp-file>
</servlet>

<servlet>
  <servlet-name>ChartSamplesDashboard</servlet-name>
  <jsp-file>/jsp/ChartSamplesDashboard.jsp</jsp-file>
</servlet>

<servlet>
  <servlet-name>SampleDrill</servlet-name>
  <jsp-file>/jsp/SampleDrill.jsp</jsp-file>
</servlet>

<servlet>
  <servlet-name>SWDashboard</servlet-name>
  <jsp-file>/jsp/SWDashboard.jsp</jsp-file>
</servlet>

<servlet>
  <servlet-name>Map</servlet-name>
  <jsp-file>/jsp/Map.jsp</jsp-file>
</servlet>

<servlet>
  <servlet-name>SampleFlashDashboard</servlet-name>
  <jsp-file>/jsp/SampleFlashDashboard.jsp</jsp-file>
</servlet>
<!-- [END SAMPLE SERVLETS] -->

```

5. Remove the sample servlet mappings in the same fashion:

```

<!-- [BEGIN SAMPLE SERVLET MAPPINGS] -->
<servlet-mapping>
  <servlet-name>UpdateSampleEmails</servlet-name>
  <url-pattern>/UpdateSampleEmails</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>Widgets</servlet-name>
  <url-pattern>/Widgets</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>SampleDashboard</servlet-name>
  <url-pattern>/SampleDashboard</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>ChartSamplesDashboard</servlet-name>
  <url-pattern>/ChartSamplesDashboard</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>SampleDrill</servlet-name>
  <url-pattern>/SampleDrill</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>SWDashboard</servlet-name>
  <url-pattern>/SWDashboard</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>Map</servlet-name>
  <url-pattern>/Map</url-pattern>

```

```

</servlet-mapping>

<servlet-mapping>
  <servlet-name>SampleFlashDashboard</servlet-name>
  <url-pattern>/SampleFlashDashboard</url-pattern>
</servlet-mapping>
<!-- [END SAMPLE SERVLET MAPPINGS] -->

```

6. Remove the **SampleDataStartupListener**:

```

<listener>
  <listener-class>org.pentaho.platform.web.http.context.HsQLdbStartupListener</
listener-class>
</listener>

```


7. Remove the **hsqldb-databases** section:

```

<!-- [BEGIN HSQLDB DATABASES] -->
  <context-param>
    <param-name>hsqldb-databases</param-name>
    <param-value>sampdata@../../data/hsqldb/sampdata</param-value>
  </context-param>
<!-- [END HSQLDB DATABASES] -->

```

8. Remove the **SystemStatusFilter** filter:

 **Note:** This is not part of the Pentaho samples; it provides error status messages that are only useful for development and testing purposes, and should be removed from a production system.

```

<filter>
  <filter-name>SystemStatusFilter</filter-name>
  <filter-class>com.pentaho.ui.servlet.SystemStatusFilter</filter-class>
  <init-param>
    <param-name>initFailurePage</param-name>
    <param-value>InitFailure</param-value>
    <description>This page is displayed if the PentahoSystem fails to properly
initialize.</description>
  </init-param>
</filter>

```

9. Remove the filter mapping for the **SystemStatusFilter**:

```

<filter-mapping>
  <filter-name>SystemStatusFilter</filter-name>
  <url-pattern>/*</url-pattern>
</filter-mapping>

```

10. Save and close the web.xml file.


11. Delete the `/pentaho/server/biserver-ee/data/` directory.

This directory contains a sample database, control scripts for that database and the environment settings it needs to run, and SQL scripts to initialize a new Pentaho solution repository.

Your BI Server instance is now cleaned of samples and development/testing pieces, and is streamlined for production.

BI Suite Graphical Installer Deployment

If you used the graphical installation utility previously, it's easiest to use archive packages to replace certain parts of the BI Suite that have changed. **There is no simple way to upgrade by using the latest BI Suite installation utility**, and presently Pentaho does not have the ability to automatically upgrade an existing installation. Therefore you will have to merge your old settings and solutions by hand. All of the necessary instructions for this process are below.

 **Note:** If you are concerned about the complexity of this upgrade process and do not feel that you can complete it on your own, you can hire a consultant to perform it for you. Contact Pentaho Services through your sales or support representative to learn more.

Graphical Installer Deployment Upgrade Checklist

The Upgrade Checklist is a concise list of instructions intended to show a high-level overview of the upgrade process. It also serves as a method of verifying that each task is performed in the correct order. You may find it useful to print the

checklist out and physically mark each step in the Done column as you complete it. **The checklist is not the complete instruction set**; consult the verbose instructions throughout this guide for more details on each step.

Step	Procedure	Done
Step 1	Download the upgrade materials from the Enterprise Edition FTP site or Pentaho Knowledge Base.	
Step 2	Stop all Pentaho programs and services, including the solution repository database.	
Step 3	Back up your /pentaho/server/ directory and hibernate and quartz databases.	
Step 4	Rename the /pentaho/server/ directory to /pentaho/server_old/, and create a new empty /pentaho/server/ directory to replace it.	
Step 5	Unpack the BI Server archive package to the empty /pentaho/server/ directory.	
Step 6	Copy any necessary JDBC drivers for your solution database and data sources to the /pentaho/server/biserver-ee/tomcat/lib/ directory.	
Step 7	Copy your old web.xml and context.xml from the old Pentaho WAR to the new one.	
Step 8	Copy over your old server.xml to the new /tomcat/conf/ directory.	
Step 9	Unpack any Pentaho User Console plugins you have licenses for (Analyzer, Dashboard Designer, Interactive Reporting) to the /pentaho/server/biserver-ee/pentaho-solutions/system/ directory.	
Step 10	If you have a Pentaho BI Suite For Hadoop license, unpack the pdi-hadoop-plugin and pdi-hadoop-plugin-ee archives to the /pentaho/server/biserver-ee/pentaho-solutions/system/kettle/plugins directory.	
Step 11	If you have a Pentaho Analysis Enterprise Edition license, unpack the pentaho-analysis-ee archive to a temporary location, then copy the requisite JARs to the /pentaho/server/biserver-ee/tomcat/webapps/pentaho/WEB-INF/lib/ directory, and the requisite configuration files to the /pentaho/server/biserver-ee/tomcat/webapps/pentaho/WEB-INF/classes/ directory.	
Step 12	Copy over the scripts directories from the old enterprise-console and tomcat directories to the new ones.	
Step 13	For 64-bit Windows systems, copy the Tomcat executables from the old BI Server instance to the new one.	
Step 14	For Windows deployments, copy over the Start Menu icons from the old BI Server instance to the new one.	
Step 15	Remove the Pentaho sample content.	
Step 16	Copy all of your solution directories from the /pentaho/server_old/biserver-ee/pentaho-solutions/ directory to the new one.	
Step 17	Copy the files documented in Merging Your BI Server Configuration Files on page 32 to the new pentaho-solutions directory.	
Step 18	Edit the following files and merge in the changes as instructed: pentahoObjects.spring.xml, mondrian.properties, and web.xml.	
Step 19	If you're a PDI Enterprise Edition customer, continue on to the section that describes the DI Server upgrade process.	
Step 20	Copy the following files from the old Pentaho Enterprise Console to the new one: console.xml, console.properties, login.properties, login.conf, log4j.xml; and the entire /resource/config/hsqldb/ directory.	
Step 21	If you have not already done so, move data source JDBC driver JARs from the old directories to the new ones. This includes the BI Server and Enterprise Console.	
Step 22	If you installed the Interactive Reporting plugin, use the license installer utility (or start the BI Server and Enterprise Console and use the license interface in Enterprise Console) to install the Pentaho Reporting Enterprise Edition license key.	
Step 23	Start the solution database, BI Server, and Pentaho Enterprise Console and check for any obvious errors or failures in the console output and log file.	
Step 24	Check that all of your content is accessible. Log into the Pentaho User Console and verify that your product licenses work. Run a new report in Interactive Reporting to ensure that it is ready for production. Verify at least two existing analysis views or Analyzer reports for each data source. Verify that any previously created dashboards are functioning properly. Run all reports that have been published from Report Designer. Lastly, check all of your schedules to ensure that they are still properly configured and operational.	

Step	Procedure	Done
Step 25	On workstations, remove old client tool directories, then unpack applicable client tools to the empty <code>/pentaho/design-tools/</code> directory.	
Step 26	Verify that all client tools can access old content and publish to the BI Server. Verify that PDI can connect to the DI Server, if applicable.	
Step 27	Remove the <code>/pentaho/server_old/</code> directory, and any archive packages and temporary directories you used for the upgrade.	

Obtaining the Installation Materials

As an Enterprise Edition customer, you can obtain the BI Server and client tool pre-configured packages from the Pentaho Enterprise Edition FTP site by using your 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.

With the exception of Design Studio, which has separate packages for Linux, OS X, and Windows, all of the below-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 admins. Other than this archive technology difference, the BI Suite packages are identical.

The Pentaho Enterprise Console is included in the BI Server package, so you do not need to download it separately.

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

Windows

The packages you will need for a Windows server are:

- **BI Server (includes the Pentaho Enterprise Console):** `/4.1.0-GA/server/windows/biserver-ee-3.10.0-GA.zip`
- **Pentaho Analyzer plugin:** `/4.1.0-GA/server/plugins/paz/`
- **Pentaho Dashboard Designer plugin:** `/4.1.0-GA/server/plugins/pdd/`
- **Pentaho Interactive Reporting plugin:** `/4.1.0-GA/server/plugins/pir/`
- **BI Server Hadoop plugin:** `/Pentaho_Data_Integration/4.2.1-GA/server/plugins/pdi-hadoop-plugin-1.0.3.zip`
- **BI Server Hadoop Enterprise Edition plugin:** `/Pentaho_Data_Integration/4.2.1-GA/server/plugins/pdi-hadoop-plugin-ee-1.0.4.zip`

The packages you will need for your Windows workstations are:

- **Report Designer:** `/4.1.0-GA/client/windows/prd-ee-4.1.0-GA.zip`
- **Metadata Editor:** `/4.1.0-GA/client/windows/pme-ee-4.1.0-GA.zip`
- ***Schema Workbench:** `/4.1.0-GA/client/windows/psw-ee-3.3.0.14682.zip`
- **Design Studio for 32-bit Windows:** `/4.1.0-GA/client/windows/pds-ee-win-32-4.0.0-GA.zip`
- **Design Studio for 64-bit Windows:** `/4.1.0-GA/client/windows/pds-ee-win-64-4.0.0-GA.zip`
- **Design Studio plugins for existing Eclipse installations:** `/4.1.0-GA/client/design-studio/org.pentaho.designstudio.editors.actionsequence_4.0.0.GA.zip`
- **Aggregation Designer:** `/4.1.0-GA/client/windows/pad-ee-1.3.0-GA.zip`
- **Data Integration:** `/4.1.0-GA/client/windows/pdi-ee-client-4.2.1-GA.zip`

Linux and Solaris

The packages you will need for a Linux or Solaris server are:

- **BI Server (includes the Pentaho Enterprise Console):** `/4.1.0-GA/server/nix/biserver-ee-3.10.0-GA.tar.gz`
- **Pentaho Analyzer plugin:** `/4.1.0-GA/server/plugins/paz/`
- **Pentaho Dashboard Designer plugin:** `/4.1.0-GA/server/plugins/pdd/`
- **Pentaho Interactive Reporting plugin:** `/4.1.0-GA/server/plugins/pir/`
- **BI Server Hadoop plugin:** `/Pentaho_Data_Integration/4.2.1-GA/server/plugins/pdi-hadoop-plugin-1.0.3.zip`
- **BI Server Hadoop Enterprise Edition plugin:** `/Pentaho_Data_Integration/4.2.1-GA/server/plugins/pdi-hadoop-plugin-ee-1.0.4.zip`

The packages you will need for your Linux or Solaris workstations are:

- **Report Designer:** /4.1.0-GA/client/nix_osx/prd-ee-4.1.0-GA.tar.gz
- **Metadata Editor:** /4.1.0-GA/client/nix_osx/pme-ee-4.1.0-GA.tar.gz
- ***Schema Workbench:** /4.1.0-GA/client/nix_osx/psw-ee-3.3.0.14682.tar.gz
- **Design Studio for 32-bit Linux:** /4.1.0-GA/client/nix_osx/pds-ee-linux-32-4.0.0-GA.tar.gz
- **Design Studio for 64-bit Linux:** /4.1.0-GA/client/nix_osx/pds-ee-linux-64-4.0.0-GA.tar.gz
- **Design Studio plugins for existing Eclipse installations:** /4.1.0-GA/client/design-studio/org.pentaho.designstudio.editors.actionsequence_4.0.0.GA.zip
- **Aggregation Designer:** /4.1.0-GA/client/nix_osx/pad-ee-1.3.0-GA.tar.gz
- **Data Integration:** /4.1.0-GA/client/nix_osx/pdi-ee-client-4.2.1-GA.tar.gz



Note: The PSW package name's exact version number may change between the publication of this list and the GA release.

Deploying the BI Server

Follow the directions below to install a new BI Server and the Pentaho Enterprise Console onto your server. The examples assume a Linux environment; adjust the paths accordingly for Windows-based machines.

1. Create a new server directory to replace the one you renamed earlier.

```
mkdir /home/pentaho/pentaho/server/
```

2. Untar or unzip the **biserver-ee-3.10.0-GA** archive from wherever you downloaded it to the new server directory.

```
tar zxvf /home/amenethil/downloads/biserver-ee-3.10.0-GA.tar.gz -C /home/pentaho/pentaho/server/
```

This will create new **biserver-ee** and **enterprise-console** subdirectories.

The BI Server and Pentaho Enterprise Console are now extracted into the proper directories. Continue on to the next several sections to install drivers, change the hostname and port if necessary, and install any plugins that you have support entitlements for.

Copying context.xml, server.xml, and web.xml

The context.xml, server.xml, and web.xml files contain important configuration information that must be transferred to your new BI Server instance. Some of these files may need to be changed later in the upgrade process, but for right now, you must copy them to the new deployment.

1. Copy your old **context.xml** from `/server_old/biserver-ee/tomcat/webapps/pentaho/META-INF/` to `/server/biserver-ee/tomcat/webapps/pentaho/META-INF/`.
2. Copy your old **web.xml** from `/server_old/biserver-ee/tomcat/webapps/pentaho/WEB-INF/` to `/server/biserver-ee/tomcat/webapps/pentaho/WEB-INF/`.
3. Copy your old **server.xml** from `/server_old/biserver-ee/tomcat/conf/` to `/server/biserver-ee/tomcat/conf/`.

Your core BI Server configuration files have been transferred to your new BI Server.

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
- Pentaho Interactive Reporting



Note: The Pentaho Analysis Enterprise Edition plugin is separate from Analyzer, and requires a different installation process explained in another part of this guide.

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**; the Interactive Reporting plugin always starts with **pir**.

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

This will create a subdirectory for each plugin that you unpack.

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. If this directory does not exist, create it.

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.

Installing the Analysis Enterprise Edition Plugin

Follow the instructions below to install the Pentaho Analysis Enterprise Edition package, which contains Analysis engine enhancements for large ROLAP deployments.



Note: This procedure does not cover Pentaho Analyzer installation.



Note: If you are performing a manual WAR build and deployment, and you want to slipstream the Pentaho Analysis Enterprise Edition JARs into the WAR pre-build, you can substitute `/biserver-manual-ee/build-resources/custom-pentaho-webapp/WEB-INF/` for `/WEB-INF/` references below.

1. If you have not already done so, retrieve the **pentaho-analysis-ee-3.3.0-GA** package from the Pentaho Enterprise Edition FTP site (login credentials were emailed to you with your Welcome Kit).
2. Unpack the archive to a temporary location.
3. If it is currently running, shut down the BI Server.
4. Copy only the following JARs from the `/pentaho-analysis-ee/lib/` directory to the `/WEB-INF/lib/` directory inside of the deployed `pentaho.war`:
 - `infinispan-core-4.2.1.FINAL`
 - `jboss-transaction-api-1.0.1.GA`
 - `jcip-annotations-1.0`
 - `jgroups-2.12.0.CR5`
 - `marshalling-api-1.2.3.GA`
 - `memcached-0.0.1-PENTAHO`
 - `pentaho-analysis-ee-3.3.0-GA-obf`
 - `rhq-pluginAnnotations-3.0.1`
 - `river-1.2.3.GA`
5. Copy all of the configuration files from `/pentaho-analysis-ee/config/` to the `/WEB-INF/classes/` directory inside of your deployed `pentaho.war`.

6. Remove the temporary **pentaho-analysis-ee** directory.

Pentaho Analysis Enterprise Edition is now installed with the default Infinispan configuration.

If you need to switch from Infinispan to Memcached, continue to the next section.

Switching to Memcached

In order to complete this procedure, you must have your own pre-configured Memcached instance. You should have also installed the Analysis Enterprise Edition package to your BI Server or standalone Mondrian engine.

If you already use the Memcached cache framework in your organization and would like to hook it up to the Pentaho Analysis ROLAP engine, follow the directions below to switch from the default Infinispan cache framework.

1. If the BI Server or standalone Mondrian engine are running, shut them down now.
2. If you performed a default install of the Pentaho Analysis Enterprise Edition package, then you should have all of the required JARs installed to the BI or Mondrian server. If you aren't sure, verify now that the following JARs are present in the `/WEB-INF/lib/` directory inside of your deployed `pentaho.war` or Mondrian engine:

- pentaho-analysis-ee
- commons-lang
- commons-io
- commons-codec
- pentaho-ee-dsc-core
- memcached

3. Edit the **pentaho-analysis-config.xml** in the `/WEB-INF/classes/` directory inside the deployed `pentaho.war` or Mondrian engine, and change the value of **SEGMENT_CACHE_IMPL** to match the class name referenced below:

```
<entry key="SEGMENT_CACHE_IMPL">com.pentaho.analysis.segmentcache.impl.memcached.MemcachedSegmentCache</entry>
```

4. Edit the **memcached-config.xml** in the `/WEB-INF/classes/` directory inside the deployed `pentaho.war` or Mondrian engine, and change the values of **SALT**, **SERVERS**, and **WEIGHT** to match your preference:

Refer to [Memcached Configuration Options](#) on page 12 for more information on these settings.

```
<entry key="SALT">YOUR SECRET SALT VALUE HERE</entry>


<entry key="SERVERS">192.168.0.1:1642,192.168.0.2:1642</entry>

<entry key="WEIGHTS">1,1</entry>
```

Your Pentaho Analysis Enterprise Edition instance is now configured to use Memcached for ROLAP segment caching.

Memcached Configuration Options

These properties control Memcached settings, and are set in the **memcached-config.xml** file in the `/WEB-INF/classes/` directory inside of your deployed `pentaho.war` or Mondrian engine.

 **Note:** This is not a comprehensive list of the potential Memcached settings; the options explained below are the ones most critical to Memcached configuration for Pentaho Analysis. You must research further Memcached options on your own.

Property	Purpose
SERVERS	A comma-separated list of servers and port numbers representing the Memcached nodes usable by the plugin.
WEIGHTS	A comma-separated list of numbers representing the relative caching capacity of the servers defined in the SERVERS property. There must be exactly as many values of WEIGHTS as there are values of SERVERS. As an example, if the first server has a capacity of 128 megabytes, and the second has a capacity of 256 megabytes, the correct values for the WEIGHTS property should be "1,2", indicating that the first server has a relative size of half of the second one.
SALT	A secret key prefix to be used when saving and loading segment data from the Memcached nodes. This property must be the same for all Mondrian nodes that share their

Property	Purpose
	caches. If the SALT value is different from one node to the next, the nodes will not be able to share their cache data.

Copying Startup Scripts

The BI Suite installer includes a series of scripts that start and stop Pentaho services. These are different between Linux and Windows, but the directories have the same names. These scripts are not included in the archive packages, so you must copy them over from various directories in your old Pentaho instance.

1. Copy your old `/server_old/biserver-ee/tomcat/scripts/` directory to the new tomcat directory.
2. Copy your old `/server_old/enterprise-console/scripts/` directory to the new enterprise-console directory.

All of the necessary BI Server startup scripts have been copied. However, if you installed the DI Server, its scripts will be copied later as part of the DI Server upgrade procedure.

Copying Windows 64-bit Tomcat Executables



Note: If you are using a 32-bit version of the BI Server, you do not need to follow this procedure.

If you are upgrading a 64-bit Windows BI Server that was deployed via the graphical installation utility, you will have to copy over the old Tomcat executable files and a batch script. The archive distribution is 32-bit only for Windows deployments; however, the version of Tomcat the Pentaho ships has not changed between BI Server 3.9 and 3.10. Therefore, you can simply copy these three files from `\pentaho\server_old\biserver-ee\tomcat\bin\` to the new bin directory:

1. tomcat6.exe
2. tomcat6w.exe
3. service.bat

If you do not copy these files, the resulting BI Server instance will be running on a 32-bit application server.

Copying Windows Start Menu Icons



Note: This procedure applies only to Windows users.

Copy all of the `.ico` files from `\pentaho\server_old\biserver-ee\` to the new biserver-ee directory. These are the icon graphics for the Windows Start menu.

Removing Pentaho Sample Data and Solutions

By default, Pentaho provides a sample data source and a solution directory filled with example content. These samples are provided for evaluation and testing. Once you are ready to move from an evaluation or testing scenario to development or production, you should remove the samples so that they do not get in the way of your BI content. Follow the instructions below to completely remove the Pentaho sample data and solutions.

1. Stop the BI Server.
2. If they exist, delete the `/pentaho/server/biserver-ee/pentaho-solutions/steel-wheels/` and `/pentaho/server/biserver-ee/pentaho-solutions/bi-developers/` directories.
These directories contain sample content for CDF and BI Server plugins. They are not present in typical manual deployment scenarios.
3. Edit the `/pentaho/WEB-INF/web.xml` file inside of the deployed pentaho.war.
As laid down by the Pentaho graphical installer and archive packages, this path should be `/pentaho/server/biserver-ee/tomcat/webapps/pentaho/WEB-INF/web.xml`. If you performed a manual WAR build and deployment, then you must adjust the path to fit your configuration.
4. Remove the sample servlets by commenting them out or removing them from the file:

```
<!-- [BEGIN SAMPLE SERVLETS] -->
<servlet>
  <servlet-name>UpdateSampleEmails</servlet-name>
  <jsp-file>/jsp/UpdateSampleEmails.jsp</jsp-file>
</servlet>
```

```

<servlet>
  <servlet-name>Widgets</servlet-name>
  <jsp-file>/jsp/Widgets.jsp</jsp-file>
</servlet>

<servlet>
  <servlet-name>SampleDashboard</servlet-name>
  <jsp-file>/jsp/SampleDashboard.jsp</jsp-file>
</servlet>

<servlet>
  <servlet-name>ChartSamplesDashboard</servlet-name>
  <jsp-file>/jsp/ChartSamplesDashboard.jsp</jsp-file>
</servlet>

<servlet>
  <servlet-name>SampleDrill</servlet-name>
  <jsp-file>/jsp/SampleDrill.jsp</jsp-file>
</servlet>

<servlet>
  <servlet-name>SWDashboard</servlet-name>
  <jsp-file>/jsp/SWDashboard.jsp</jsp-file>
</servlet>

<servlet>
  <servlet-name>Map</servlet-name>
  <jsp-file>/jsp/Map.jsp</jsp-file>
</servlet>

<servlet>
  <servlet-name>SampleFlashDashboard</servlet-name>
  <jsp-file>/jsp/SampleFlashDashboard.jsp</jsp-file>
</servlet>
<!-- [END SAMPLE SERVLETS] -->

```

5. Remove the sample servlet mappings in the same fashion:

```

<!-- [BEGIN SAMPLE SERVLET MAPPINGS] -->
<servlet-mapping>
  <servlet-name>UpdateSampleEmails</servlet-name>
  <url-pattern>/UpdateSampleEmails</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>Widgets</servlet-name>
  <url-pattern>/Widgets</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>SampleDashboard</servlet-name>
  <url-pattern>/SampleDashboard</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>ChartSamplesDashboard</servlet-name>
  <url-pattern>/ChartSamplesDashboard</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>SampleDrill</servlet-name>
  <url-pattern>/SampleDrill</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>SWDashboard</servlet-name>
  <url-pattern>/SWDashboard</url-pattern>
</servlet-mapping>

```

```

<servlet-mapping>
  <servlet-name>Map</servlet-name>
  <url-pattern>/Map</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>SampleFlashDashboard</servlet-name>
  <url-pattern>/SampleFlashDashboard</url-pattern>
</servlet-mapping>
<!-- [END SAMPLE SERVLET MAPPINGS] -->

```

6. Remove the `SampleDataStartupListener`:

```

<listener>
  <listener-class>org.pentaho.platform.web.http.context.HsqldbStartupListener</
listener-class>
</listener>

```

7. Remove the `hsqldb-databases` section:

```

<!-- [BEGIN HSQLDB DATABASES] -->
  <context-param>
    <param-name>hsqldb-databases</param-name>
    <param-value>sampladata@../..../data/hsqldb/sampladata</param-value>
  </context-param>
<!-- [END HSQLDB DATABASES] -->

```

8. Remove the `SystemStatusFilter` filter:



Note: This is not part of the Pentaho samples; it provides error status messages that are only useful for development and testing purposes, and should be removed from a production system.

```

<filter>
  <filter-name>SystemStatusFilter</filter-name>
  <filter-class>com.pentaho.ui.servlet.SystemStatusFilter</filter-class>
  <init-param>
    <param-name>initFailurePage</param-name>
    <param-value>InitFailure</param-value>
    <description>This page is displayed if the PentahoSystem fails to properly
initialize.</description>
  </init-param>
</filter>

```

9. Remove the filter mapping for the `SystemStatusFilter`:

```

<filter-mapping>
  <filter-name>SystemStatusFilter</filter-name>
  <url-pattern>/*</url-pattern>
</filter-mapping>

```

10. Save and close the `web.xml` file.

11. Delete the `/pentaho/server/biserver-ee/data/` directory.

This directory contains a sample database, control scripts for that database and the environment settings it needs to run, and SQL scripts to initialize a new Pentaho solution repository.

Your BI Server instance is now cleaned of samples and development/testing pieces, and is streamlined for production.

Migrating Your Solutions

BI Server content is stored in the `/pentaho/server_old/biserver-ee/pentaho-solutions/` directory. However, this is also where the BI Server's configuration files are stored, as well as Pentaho-supplied sample content. You must move your BI content from your old solutions directory to the new one. Copying the settings and other configuration information will happen in the next section.



Note: You must use the archive flag or option when you copy if you want a minimal impact on your solution database. If you copy without the archive option, all of the dates on your solution files and directories will reset to today's date, which will force the solution repository to refresh.

When you copy your solution directories, which are stored in `/pentaho/server_old/biserver-ee/pentaho-solutions/`, this should include all reports, data sources, metadata definitions, dashboards, charts, and action sequences you've created and published. **Do not copy** any other directories at this time.



Danger: Be sure that you transfer **all** of the solution directories that you've created. Some people may only have one solution directory, and some may have many directories to segregate content for different departments or data sources. If you fail to copy over a solution directory, the content it contains will not be carried over to your new Pentaho instance.

After you have copied your solutions to your new solutions directory, proceed to the next section to learn how to migrate your settings.

Merging Your BI Server Configuration Files

The following configuration files have changed between version 3.9 and 3.10:

- pentahoObjects.spring.xml
- mondrian.properties
- web.xml

Instructions for merging changes into these files are in the next subsection. Before you get to that point, however, there are a few customized configuration files that you must copy over from your old pentaho-solutions directory to the new one. These files cover security, solution repository, and audit log configuration. While they have not changed in version 3.10, they are customized according to your BI Server settings.

Copy the following **files** -- **but not any subdirectories** -- from your old BI Server `/pentaho-solutions/` directory to the new one:

- `/admin/audit/*`
- `/system/*`
- `/system/olap/datasources.xml`
- `/system/mondrian/mondrian.properties`
- `/system/google/googlesettings.xml` (if you are using your own Google Maps API key)

Copy the following **directories** and all of the files and subdirectories therein from your old BI Server `/pentaho-solutions/` directory to the new one:

- `/system/content/`
- `/system/dialects/`
- `/system/hibernate/`
- `/system/quartz/`
- `/system/metadata/`

These directories contain Pentaho samples. If they exist in your current BI Server installation, you should replace them with the new versions. If they do not exist in the old dir, you can safely delete them from the new pentaho-solutions directory:

- `/bi-developers/`
- `/steel-wheels/`

After you've copied all of these files to the proper locations in your new pentaho-solutions directory, continue on to the next subsection to learn how to merge the other files.

Updating pentahoObjects.spring.xml

Follow the instructions below to bring your old pentahoObjects.spring.xml file up to date for the BI Server 3.10.

1. Open your recently copied `/pentaho-solutions/system/pentahoObjects.spring.xml` file with a text editor.
2. Find `IMetadataDomainRepository` bean declaration, and replace it with this:


```
<!-- Concrete implementation of IMetadataDomainRepository -->
<bean id="IMetadataDomainRepositoryImpl"
  class="org.pentaho.platform.plugin.services.metadata.SecurityAwareMetadataDomainRepository"
  scope="singleton"/>
<!-- Wrap the concrete IMetadataDomainRepository implementation with one that caches
domains per session -->
<bean id="IMetadataDomainRepository"
  class="org.pentaho.platform.plugin.services.metadata.SessionCachingMetadataDomainRepository"
  scope="singleton">
  <constructor-arg><ref bean="IMetadataDomainRepositoryImpl"/></constructor-arg>
</bean>
```

3. Save the file and close the text editor.

Your pentahoObjects.spring.xml file is now brought up to modern standards.

Updating web.xml

Follow the instructions below to bring your old **web.xml** file up to date for the BI Server 3.10.

 **Note:** You may have already completed this task if you performed a manual deployment upgrade and merged your old web.xml with the new one before the build.

1. Inside of your deployed **pentaho.war** file, open the `/WEB-INF/web.xml` file with a text editor.

With an archive or graphical deployment, the path is: `/pentaho/server/biserver-ee/tomcat/webapps/pentaho/WEB-INF/`.

2. Remove the following block of XML just before the Proxy Trusting **<filter>** node:

```
<filter>
  <filter-name>DscFilter</filter-name>
  <filter-class>com.pentaho.ui.servlet.DscFilter</filter-class>
  <init-param>
    <param-name>DscFailurePage</param-name>
    <param-value>DscFailure</param-value>
    <description>This page is displayed if no BI server license is installed.</description>
  </init-param>
</filter>
```

3. Remove the following block of XML just before the **SchedulerAdminService <filter-mapping>** node:

```
<filter-mapping>
  <filter-name>DscFilter</filter-name>
  <url-pattern>/Login</url-pattern>
</filter-mapping>
```

4. Remove the following block of XML just before the **UserContent <servlet>** node:

```
<servlet>
  <servlet-name>DscFailure</servlet-name>
  <jsp-file>/jsp/DscFailure.jsp</jsp-file>
</servlet>
```

5. Remove the following block of XML just before the **UserContent <servlet-mapping>** node:


```
<servlet-mapping>
  <servlet-name>DscFailure</servlet-name>
  <url-pattern>/DscFailure</url-pattern>
</servlet-mapping>
```

6. Save the file and close the text editor.

Your web.xml file is now up to date.

Updating mondrian.properties

Follow the directions below to add new configuration options to your Pentaho Analysis engine configuration file.

 **Note:** These changes only affect Pentaho Analysis Enterprise Edition. If you don't have a license for this, you will see no benefit from modifying these settings. However, there is no harm in making the changes, especially if you plan to acquire an Analysis EE license in the future.

1. Open your recently-copied `/pentaho-solutions/system/mondrian/mondrian.properties` file and set the following two settings to **true** (or add them if they are missing):

```
mondrian.native.topcount.enable=false
```

```
mondrian.native.nonempty.enable=false
```

2. On another line in the file, add the following new Mondrian engine settings to the file:

```
# enable grouping sets, if the dialect supports it.
```

```
mondrian.rolap.groupingsets.enable=true
```

```
# Enable the datasource resolver for the Pentaho BI Platform
```

```
mondrian.spi.dataSourceResolverClass=org.pentaho.platform.web.servlet.PentahoDataSourceReso
```

3. Save and close the file.

Your mondrian.properties file is now upgraded to work with the BI Server 3.10.

Upgrading Your Single-Sign On Configuration

This only applies to people who have modified the BI Server to work with a Central Authentication Service.

Follow the directions below to enable single sign-on in your newly upgraded BI Server.

1. Unpack your deployed Pentaho WAR file in place.

Tomcat will typically do this automatically when you start it. JBoss will not typically unpack the WAR on its own, but if you unpack it by hand, the directory must have a .war extension.

2. Open a terminal or command prompt window and navigate to the `/biserver-manual-ee/build-resources/pentaho-ssso/` directory.

3. Edit the **sso-replacements.properties** file and change the default options to match your CAS configuration.

Refer to the CAS properties reference below if you have any trouble figuring out what each property does. If you have the sso-replacements.properties file that you used to enable CAS support in the BI Suite 4.0.0, you can reuse it for this process, but you must ensure that the **cas.ticket.validator.url** property is updated according to the property reference below.

4. Use Ant to run the **sso-replacements** script with the **sso-pentaho** switch, as in the following example:

```
ant -f sso-replacements.xml sso-pentaho
```

Your upgraded SSO files are now in place.

CAS Property Reference

CAS Properties

These properties mostly refer to CAS services:

Property	Description	Example
cas.authn.provider	Required. Security back-end that CAS should use. Valid values are memory, jdbc, or ldap	ldap
cas.login.url	Required. CAS login URL.	<code>\${cas.base.url}/login</code>
cas.ticket.validator.url	Required. CAS ticket validator URL.	<code>\${cas.base.url}</code>
cas.logout.url	Required. CAS logout URL. A <code>service.logout.url</code> will be appended to this URL.	<code>\${cas.base.url}/logout?url=</code>
cas.base.url	URL under which all CAS services reside.	<code>https://localhost:8443/cas</code>

Pentaho Properties

These are service URLs that serve as callbacks from the CAS server into the BI Platform:

Property	Description	Example
pentaho.service.url	Required. Processes CAS callback.	<code>\${_pentaho.service.base.url}/j_spring_cas_security_check</code>
pentaho.service.logout.url	Required. URL to go to after CAS logout.	<code>\${_pentaho.service.base.url}/Home</code>
pentaho.service.solutions.system.dir	Path to pentaho-solutions/system.	<code>/usr/local/pentaho/server/biserver-ee/pentaho-solutions/system/</code>
pentaho.service.lib.dir	Path to webapp lib directory.	<code>/usr/local/tomcat/common/lib/</code>
pentaho.service.web.xml	Path (including filename) of webapp's web.xml.	<code>/usr/local/tomcat/conf/web.xml</code>
pentaho.service.appctx.cas.xml	Path (including filename) of new applicationContext-spring-security-cas.xml.	<code>/usr/local/pentaho/server/biserver-ee/pentaho-solutions/system/applicationContext-spring-security-cas.xml</code>

Property	Description	Example
pentaho.service.jsp.dir	Path to directory containing webapp's JSPs.	/usr/local/tomcat/webapps/pentaho/jsp/
pentaho.service.spring.beans.xml	Path (including filename) of pentaho-spring-beans.xml.	/usr/local/pentaho/server/biserver-ee/pentaho-solutions/system/pentaho-spring-beans.xml
pentaho.service.base.url	Service base URL.	http://localhost:8080/pentaho
pentaho.service.pentaho.war.dir	Webapp exploded WAR directory.	/usr/local/tomcat/webapps/pentaho/
pentaho.service.webinf.dir	Path to webapp's WEB-INF directory.	/usr/local/tomcat/webapps/pentaho/WEB-INF/

Upgrading the Pentaho Enterprise Console

The new BI Server will not work properly without upgrading the Pentaho Enterprise Console. To upgrade, follow the below process.

1. If you used the manual deployment process, you must now unpack the `pec-4.1.0-GA` zip or tar.gz file to the `/pentaho/server/` directory.

This will create a new **enterprise-console** directory with default settings.



Note: If you took the archive-based or graphical installer upgrade paths, you do not need to retrieve a new Enterprise Console package; it was included in the BI Server archive package. This step is only for people who have performed a manual WAR deployment.

2. Copy the following files from your old `/enterprise-console/resource/config/` directory into the new one:

- `console.xml`
- `console.properties`
- `login.properties`
- `login.conf`
- `log4j.xml`

```
cp /home/pentaho/pentaho/server_old/enterprise-console/resource/config/console.* /  
home/pentaho/pentaho/server/enterprise-console/resource/config/ && cp /home/pentaho/  
pentaho/server_old/enterprise-console/resource/config/log* /home/pentaho/pentaho/  
server/enterprise-console/resource/config/
```


3. Copy the your old `/enterprise-console/resource/config/hsqldb/` directory to the new Enterprise Console instance, overwriting the files that are there.


This directory contains PDI-specific settings.

```
cp -r /home/pentaho/pentaho/server_old/enterprise-console/resource/config/hsqldb/ /  
home/pentaho/pentaho/server/enterprise-console/resource/config/
```


The Pentaho Enterprise Console has been upgraded to version 4.1.0.

Moving JDBC Drivers

 **Note:** JDBC driver versions prior to 4 may have issues with Java 6. Read your database vendor's compatibility notes carefully before moving or downloading driver JARs.

 **Note:** Microsoft SQL Server customers frequently use an alternative, non-vendor-supported driver called JTDS. Ensure that you are downloading the expected driver before installing it.

In order for your existing data source connections to work, you must copy over your old JDBC driver JARs for all of your data sources. The relevant driver directories are below:

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

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

Testing Your New BI Server

Your server is now ready to test. From an end-user workstation, start a Web browser and navigate to **http://example.com:8080/pentaho** (change **example.com** to your BI Server's hostname, domain name, or IP address). You should see a screen like the picture below:



Click the **Login** button, log in normally with an existing user account, and verify that you can run all of your old BI content. Also verify that your schedules are still in place and functional. Specifically, Pentaho recommends that you verify the following actions if they apply to your Enterprise Edition scenario:

1. Open a report that requires no user prompts.
2. Open a report that requires user prompting (a parameterized report).
3. If you are a Pentaho Reporting Enterprise Edition customer, create a report using Interactive Reporting.
4. If you are a Pentaho Analysis Enterprise Edition customer, create a new Analyzer report.
5. If you are a Pentaho Reporting Enterprise Edition customer, create a new report through Interactive Reporting.
6. Open a dashboard if you have a Dashboard Designer support entitlement.
7. Create a new chart using Chart Designer if you have a Dashboard Designer support entitlement.
8. Publish a report from Report Designer.
9. Publish an analysis schema from Spoon.
10. After all other tests have been performed, check your server status in the Pentaho Enterprise Console.
11. Check your application server log for any errors.

Upgrading the Pentaho Client Tools

Upgrading the Pentaho client tools on each user's workstation is considerably easier than upgrading the BI Server. You can either download and run the BI Suite installer, or download and unpack individual client tool archive packages. If you choose the installer option, simply rename or remove the old client tools, then run the installer, un-check the default options, select the client tools you want to install, and proceed through the rest of the installation. If you choose the archive package option, simply remove or rename the old client tool's directory, then unpack the new archive to the original location.

The only Pentaho client tools that store user settings are Report Designer and Pentaho Data Integration (Spoon). When upgrading these programs, the new instance will detect the custom settings from the old instance and use them automatically. These settings for Report Designer are stored in the **/.pentaho** directory, and the settings are stored in the **/.kettle** directory, both in each user's home or user directory. No settings or content files are stored in the `/pentaho/design-tools/report-designer/` or `/pentaho/design-tools/data-integration/` directories (or the subdirectories for any other client tools). This means that you can safely remove the old Report Designer and Data Integration instances without harming the custom user settings.

Post-Upgrade Configuration

Once your upgrade is complete, the only configuration you have to do is in regards to any new software you have installed. If you installed any new BI Server plugins, such as:

- Dashboard Designer
- Analyzer or Pentaho Analysis Enterprise Edition
- Interactive Reporting
- Hadoop (or Hive metadata support)

Or any new PDI plugins related to Hadoop, then you will have to install the appropriate license keys in the same manner that you previously installed your BI Suite licenses.



Note: You do not need to reinstall any extant valid license keys.

Cleanup

Now that your new BI Server version 4.1 is built, deployed, and tested, you can clean up some old or temporary files. Typically this would include:

- The temporary **biserver-manual-ee** or **biserver-ee** (from the archive package) directory
- Your **server_old** directory
- Archive packages for client tools, servers, and plugins
- 4.0 backup artifacts, though you may want to keep these for a while, in case you need to roll back

Troubleshooting

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

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.

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.

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.

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.

Tomcat Logs Report Memory Leaks

When shutting down Tomcat, you may see some SEVERE-level warnings in the log file similar to these:

```
Dec 17, 2010 10:18:19 AM org.apache.catalina.loader.WebappClassLoader
clearReferencesJdbc
SEVERE: The web application [/pentaho] registered the JDBC driver
[mondrian.olap4j.MondrianOlap4jDriver] but failed to unregister it when the web
application was stopped. To prevent a memory leak, the JDBC Driver has been forcibly
unregistered.
Dec 17, 2010 10:18:19 AM org.apache.catalina.loader.WebappClassLoader
clearReferencesThreads
SEVERE: The web application [/pentaho] appears to have started a thread named [HSQLDB
Timer @49cf9f] but has failed to stop it. This is very likely to create a memory leak.
Dec 17, 2010 10:18:19 AM org.apache.catalina.loader.WebappClassLoader
clearReferencesThreads
SEVERE: The web application [/pentaho] appears to have started a thread named [MySQL
Statement Cancellation Timer] but has failed to stop it. This is very likely to create
a memory leak.
Dec 17, 2010 10:18:19 AM org.apache.catalina.loader.WebappClassLoader
clearThreadLocalMap
SEVERE: The web application [/pentaho] created a ThreadLocal
with key of type [java.lang.InheritableThreadLocal] (value
[java.lang.InheritableThreadLocal@a1320e]) and a value of type
[org.pentaho.platform.engine.security.session.TrustedSystemStartupSession] (value
[org.pentaho.platform.engine.security.session.TrustedSystemStartupSession@111089b]) but
failed to remove it when the web application was stopped. This is very likely to create
a memory leak.
```

These warnings are nothing to be concerned about when shutting down the Tomcat server, since they report problems with processes that are immanently being killed. However, they can have significance if you are only restarting or redeploying the Pentaho BI Server or DI Server Web applications. To avoid any memory leak problems in redeployment, you should restart Tomcat instead of redeploying or restarting the Web application with a live server.