Upgrade DI Systems

Help and Support Resources

If you do not find answers to your quesions here, please contact your Pentaho technical support representative.

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

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

For information about instructor-led training, visit http://www.pentaho.com/training.

Liability Limits and Warranty Disclaimer

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

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

Trademarks

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.

Third-Party Open Source Software

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

Contact Us

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

Sales Inquiries: sales@pentaho.com

Contents

DI Upgrade Introduction

This section explains how to upgrade version 4.4.x of the Pentaho Data Integration (DI) Server, plugins, and design tools to version 5.0. Version 5.0 contains major usability, stability, and architectural improvements. This version also includes many bug fixes and new features, as well as enhanced support for Big Data.

Note: To upgrade the Pentaho Business Analytics (BA) Server, plugins, and design tools see *Upgrade BA Systems*.

We recommend that you read through all upgrade documentation before you perform tasks. Familiarity with the entire process can help you plan your strategy and increase the likelihood of a successful upgrade experience.

Note: If you have a pre-4.4.x version of DI Server and you want to upgrade to version 5.0, you must do so incrementally, version by version, until you reach version 4.4.x. Upgrade instructions appear in the documentation for each version of the software. After you have upgraded to version 4.4.x, you can use these instructions to migrate to version 5.0.

Prerequisites

You need a working version of DI Server version 4.4.x.

Expertise

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

Tools

You need these tools to perform upgrade tasks.

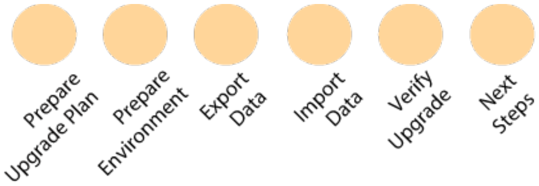
- Pentaho-supplied Migration Tool. Instructions for how to obtain this tool appear in the *Prepare Environment* section.
- File Difference tool of your choice.
- Zip or Archive tool of your choice.
- Text editor of your choice.

Note: These instructions are only for Enterprise Edition upgrades. You cannot upgrade a PDI 4.4 Community Edition deployment to PDI 5.0 Enterprise Edition through this process.

Login Credentials

You need access to an account that has administrative privileges. Additionally, Linux users might need to use the **root** account for some tasks.

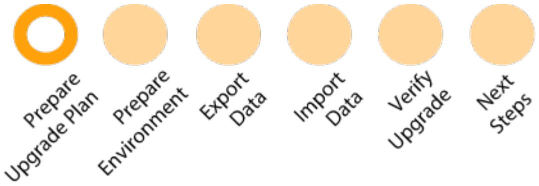
DI Upgrade Overview



The upgrade process consists of the steps indicated in the *guidepost*. Each step consists of one or more tasks.

- *Prepare Upgrade Plan*: Prepare your upgrade strategy.
- *Prepare Environment*: Install the new version of the server, download and unpack the Migration Tool.
- *Export Data From Old Server*: Use the Migration Tool to export data from the old server to migration bundle file.
- *Import Data To New Server*: Use the Migration Tool to import migration bundle file to the new server. Manually migrate additional data, as needed.
- *Verify Upgrade*: Use checklists to test that upgrades were completed properly.
- *Next Steps*: Indicates what to do after the upgrade process is complete.

Prepare Upgrade Plan



Planning essential for a successful upgrade experience. As you formulate your strategy, consider these questions.

- *Should You Upgrade To This Version?*
- *What Resources Do You Need?*
- *When Should You Upgrade?*
- *How Much Time Do You Need?*
- *What Data Will Be Migrated?*
- *Where Are The Components In The Old System Located?*
- *Where Will The Components In The New System Be Located?*
- *How Will You Prepare Users For The Upgrade Process?*
- *How Will You Document Your Upgrade Plan?*

Should You Upgrade To Version 5.0?

As you determine whether to upgrade your system to this version of the software, consider these questions.

- Does this version of the software contain new features or bug fixes that your organization needs? Review *Explore What's New for Pentaho Data Integration* for more details.
- Is your current system near or at End of Life? See the Support section of the *Pentaho Website* to determine if your version of the system is still being supported.

What Resources Do You Need?

Resources are a key component in an upgrade plan. Consider both technical and human resources.

- Do your systems meet the hardware and software requirements indicated in the *Supported Components* matrix?
- Who should be involved in the upgrade process? Smaller systems that are housed on a single computer can sometimes be upgraded by one person. But, larger or geographically distributed systems might involve more people. For example, the web application server and security might be handled by one group while reporting databases might be handled by another. List those who can provide needed information or help troubleshoot problems. Remember to include stakeholders.
- Who will help verify that data has been migrated correctly? Keep in mind that some information, such as user favorites, are sometimes best verified by end users.
- You will need to download the Migration Tool from our site. You also need a text editor, and a tool to compare the differences between text files.

How Much Time Will The Upgrade Process Take?

The amount of time it takes to perform the upgrade process varies based on how much data you have and how much you have configured the old system. Allot time to install and configure the licenses for the new system, run the Migration Tool, perform manual configuration tasks, and to test and troubleshoot the system. After testing, allow time to configure the new system, deprovision the old server, and provision the new one. Use these time estimates as a rough guide.

Table 1: Upgrade Process Time Estimates

Task	Time Estimate	Description
Installing new system, licenses, and the Migration Tool	1 to 1 1/2 Hours	If you decide to use the installation wizard to new server, allot about one hour to run the installation wizard and the license, and to also download and unpack the Migration Tool. If you need to customize your DI Repository database, web application server, or both, set aside anywhere from an hour to an hour and a half to work through the custom installation

Task	Time Estimate	Description
		procedures, install licenses, and to download and unpack the Migration Tool.
Run Migration Tool	TBD.	*TBD.*
Verify Migration	1 Hour to Several Days	Plan to spend anywhere from a few hours to several days for validation testing. The amount of time that you spend depends on your content, the numbers of transformations, shared objects, and jobs that you have, and how exhaustively you want to test the system.

When Should You Upgrade?

Once you have identified your resources and have estimated how much time it might take to upgrade your system, consider appropriate timeframes for upgrade.

- Some parts of the process require that the old and new servers be offline. How much downtime does your organization allow?
- Consider performing the migration process during off-peak hours.
- Consider organizational events and seasonal reporting needs as you select a timeframe.

What Data Will Be Migrated?

It is important to understand what types of data is migrated from the old system to the new one. Much of the data from the old server can be migrated automatically using the Migration Tool, but some must be migrated manually. The Migration Tool is a standalone utility that you can download from our website. If you use the Migration Tool, it will attempt to migrate all of the data in the table. You cannot choose to exclude.

Table 2: Data Migrated by the Migration Tool

Type of Data	Data Migrated with the Migration Tool	Exceptions
Solution Content	*All solution content in the* `pentaho-solutions` *directory. This includes reports, properties, graphics , xactions, as well as XML files. Localization files for directories and content files. Permission settings for the solution content files*.	*The following subdirectories that are in the* `pentaho-solution` *directory:* `Admin`*,* `Systems`*, and* `Temp`*. Any Pentaho system localization data.*
Transformation and Jobs	All versions of transformations and jobs, all comments related to each version. This includes all KTR and KJB files.	
Shareable Objects	All enterprise repository shareable objects and associated connection information.	File based shareable objects that are stored in the shared.xml file, database shareable objects. Only the latest version of the shareable objects are migrated.
Schedules		
System Configuration	*Information in these files: applicationContext-security-ldap.properties, applicationContext-spring-security-hibernate.properties, pentaho.xml, quartz.properties, mondrian.properties, settings.xml for various plugins, analyzer.properties, email_config.xml. Also the following tomcat files: web.xml, context.xml, classic-engine.properties, logging.properties, log4j.xml*	*JNDI data sources named "Hibernate" or "Quartz" will not be migrated. Only the following web.xml settings are migrated: Solution-path, Base-url, Fully-qualified-server-url, Locale-language, contextClass, contextConfigLocation, pentahoObjectFactory, encoding, hsqldb-databases, Proxy Trusting Filter, TrustedIpAddrs. No custom servlets or filters will be migrated.*
User Preference Settings	*Recent files list, global settings, and whether to show the tooltips in the User Console. If you have*	

Type of Data	Data Migrated with the Migration Tool	Exceptions
	set up favorites in Mobile, favorites are migrated and appear in the Pentaho User Console.	
Users and Roles	*Usernames, passwords, username description field information, role names, role descriptions, user role assignment*	This data is only migrated if the system is using Pentaho Security to authenticate users. If not, see the Configuration section for additional detail. Note that role permissions are not migrated, regardless of authentication method, and must be manually reset.

Data that must be migrated manually includes:

- *JDBC Drivers for your data source.*
- *Memory settings. These are set during the configuration step.*
- *Changes made to security files, pentaho.xml, web.xml, and context.xml.*
- *Any additional JSPs, servlets, custom libraries, plugins, java API extensions that require a change to the pentaho-objects-spring.xml file, themes, or other resources integrated as part of the custom pentaho system.*
- *Pentaho system localization information.*
- *Generated content for deleted or otherwise inactive users.*

Where Are The Components In The Old System Located?

Identify the paths to the these components so that you can supply them when you export data from the old server.

- *Web Application Server*
- *DI Repository Database*
- *Any additional servlets, and files needed to customize your system.*

Obtain the location of design tools that are not on the server.

Where Will The Components In The New System Be Located?

Identify the paths to the these components so that you can supply them when you import data into the new server.

- *Web Application Server*
- *DI Repository Database*

Additionally, obtain the location of design tools that are not on the server.

Note: After you migrate the data from the old server into the new one, you must verify that the data in both systems match. This is easier to do if you can see the user interfaces for the old and new versions of the server at the same time.

How Will You Prepare Users For The Upgrade?

Consider how you will communicate with users, obtain feedback, and provide training.

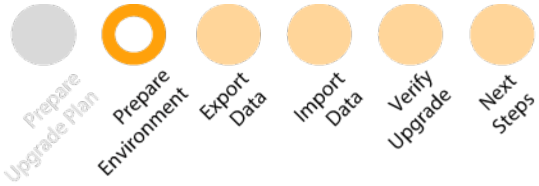
- How will you handle customer feedback about the upgrade process?
- How you will communicate with internal and external customers?
- How much advanced notice will you provide?
- How will you train users? Note that tutorials and documentation are available in the Infocenter. Additionally, Pentaho offers professionally-led training classes on many aspects of the new system. See the Pentaho website for details.

How Will You Document Your Upgrade Plan?

Documenting your responses to the questions raised in this section can be an invaluable troubleshooting aid should you need to contact technical support.

- How will you document upgrade plan?
- Who will have access to the plan and where will it be located? How will this plan be updated throughout the process?

Prepare Environment



After you have *prepared your upgrade plan*, prepare your environment. To prepare your environment, do these things.

- *Install New DI Server and Licenses*
- *Download and Install Migration Tool*

Install New DI Server and Licenses

Complete these steps to install the new DI Server and licenses.

1. If you are going to instsall the new DI Server on the same machine as the old one, make sure the old version of DI Server is running as you complete these instructions. If you decide to run the Installation Wizard, the software will sense that another system is running and will make port assignments that will enable both systems to run simultaneously. This enables you to use both systems to verify that data has migrated properly near the end of the process.
2. Install the new version of the DI Server. See *Select DI Installation Options* to determine the best installation option. Make sure that you install the samples as well.
3. Ensure that *license keys* have been installed. Do not configure the system further, just install the license keys.
4. Start the DI Server and log in as the admin user. Make sure to clear your cache. Do not configure the DI Server or make any additional changes.

   Note: If you have problems with starting the DI Server, be sure to fix it before proceeding further.

Download and Install the Migration Tool

The Migration Tool automates the data migration process. Download the Migration Tool from the Pentaho Customer Support Portal. (TBD. Will there be a CE only version of this tool made available?)

1. Download the migrator.zip file from the Customer Support Portal on the new version of the server, then unzip the file.
2. If you use Windows, open the **Command Prompt** window. If you use Linux, open the **Terminal** window.
3. Install the Migration Tool by doing one of the following.

   a) If you use Windows, enter this command.

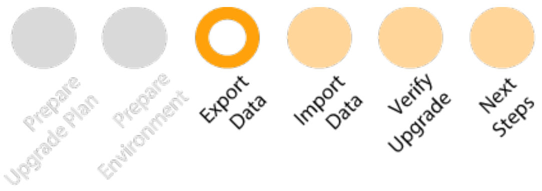
   ```
   install.bat
   ```

   b) If you use Linux, enter this command.

   ```
   ./install.sh
   ```

4. Read the EULA that appears. **Select I accept the terms of this license agreement**, then click **Next**.
5. Indicate where you want the file to be unpacked. It doesn't matter where you download the file. Click the **Next** button.
6. The **Installation Progress** window appears. Progress bars indicate the status of the process. When the process is complete, click **Quit** to exit.
7. If the old version of the server is installed on another computer, repeat these steps to install the Migration Tool on that computer.

Export Data from Old Server



After you have *prepared your environment*, export data from the old version of the server. Perform these tasks.

- *Stop DI Server and Web Application Server*
- *Ensure Old DI Repository Database is Running*
- *Use the Migration Tool to Export Data*
- *Verify Migration Export Process*

Stop Old Version of the Server

Stop the old version of the server, as well as the web application server.

Ensure Old DI Repository Database Is Running

Ensure the database that houses the old version of the DI Respository is running. If it is not, start it.

Use Migration Tool to Export Data From Old Version of the Server

Use the Migration Tool to export solution and generated content, data sources, schedules, user preferences, as well as users and roles from the old server. The Prepare Upgrade Plan step that the Migration Tool exports. Exported data is saved in a migration bundle file. A log file that contains additional details about the extraction process is also stored in the migration bundle file. After you complete the tasks in this section, you will import this migration bundle file. Instructions for that appear in the *next section* of this documentation.

To use the Migration Tool to import data, complete these steps.

1. If you have not done so already, identify the paths to the pentaho-solutions diretory, the pentaho.war file, as well as paths to additional servlets or jar files that you want to export.
2. Open a **Terminal** or **Command Prompt** window, then navigate to the directory where the Migration Tool is installed.
3. Issue a command in the **Terminal** or **Command Prompt** windows that indicates the action you want the Migration Tool to perform, as well as the locations to the pentaho-solutions directory, the pentaho.war file, and optionally any additional servlets and jar files that you want to migrate. Also indicate where you want the Migration Tool to place the migration bundle file.

   The Migration Tool command contains these parameters:

   - **--export:** Indicates this is an export command. This parameter is required if you want to export a migration bundle file.
   - **--migration-bundle-file:** Specifies the directory where you want the migration file to be stored. This parameter is optional. If you do not supply it, the migration bundle file is stored in the same directory where the Migration Tool is installed.
   - **--pentaho-solutions-dir:** Indicates the path to the `pentaho-solutions` directory for the old version of the DI Server. This parameter is required.
   - **--pentaho-war:** Indicates the path to `pentaho.war` file for the old version of the DI Server. This parameter is required.
   - **--extra-paths:** Specifies the path(s) to servlets, JSPs, JDBC drivers, or other additional files that you want migrated from the old version of the DI Server. This parameter is optional.

   a) **Windows:** Enter a migration command that follows this format in the **Command Prompt** window that uses the parameters described at the beginning of this step. The command should be formatted like this example.

Command Example:

```
migrator.bat --export --migration-bundle-file=C:\Users\pentaho\migration.zip --
pentaho-solutions-dir=C:\Users\pentaho\biserver-4.8\biserver-ee\pentaho-solutions
 --pentaho-war=C:\Users\pentaho\biserver-4.8\biserver-ee\tomcat\webapps\pentaho
 --extra-paths=C:Users\pentaho\biserver-4.8\biserver-ee\company-reports\,\Users
\pentaho\mycompany\biserver-ee\company-libs\company.jar
```

b) **Linux:** Enter a command that follows this format in the **Terminal** window that uses the parameters described at the beginning of this step. The command should be formatted like this example.
Command Example:

```
./migrator.sh --export --migration-bundle-file=/Users/mycompany/migration.zip --
pentaho-solutions-dir=/Users/mycompany/biserver-4.8/biserver-ee/pentaho-solutions
 --pentaho-war=/Users/mycompany/biserver-4.8/biserver-ee/tomcat/webapps/pentaho
 --extra-paths=/Users/mycompany/biserver-ee/company-reports/,/Users/mycompany/
biserver-ee/company-libs/company.jar
```

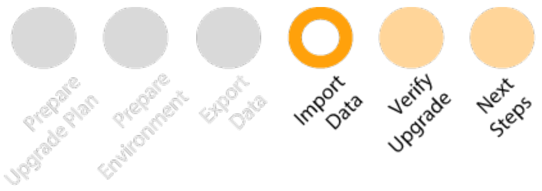
4. The Migrator Tool runs. The progress of the tool is displayed in the **Command Prompt** or **Terminal** window.

Verify Export Process

To verify that the migration bundle has been created properly and that data was extracted, do these things.

1. Open a **Terminal** or **Command Prompt** window and navigate to the directory where the migration bundle file is stored.
2. Verify that the bundle exists and that the file size is not zero.
3. Navigate to the directory where the Migration Tool was installed. The `migrator.log` file should appear. The `migrator.log` file contains a log of the export process.
4. Open the `migrator.log` file and note any failures or errors. If any failures or errors exist, you might need to migrate that data manually.
5. Close the file, then rename the `migrator.log` file to something else. If you do not, the migrator.log file will be overwritten when you perform the tasks in the *Import Data to New Server* step.

Import Data to New Server



After you have *exported your data from the old server and into the migration bundle file*, import the data in the migration bundle file into the new server. Perform these tasks.

- *Stop DI Server and Web Application Server*
- *Ensure New DI Repository Database is Running*
- *Use the Migration Tool to Import Data*
- *Verify Migration Import Process*
- *Manually Migrate Additional Data*

Stop New Version of DI Server

Stop the old version of the server, as well as the web application server.

Ensure New DI Repository Database Is Running

Ensure the database that houses the new version of the DI Respository is running. If it is not, *start it*.

Use Migration Tool to Import Data to New Server

Use the Migration Tool to import the migration bundle file to the new server. The migration bundle file was generated in the *previous step*. To use the Migration Tool to import data, complete these steps.

1. If you have not done so already, identify the paths to the pentaho-solutions diretory, the pentaho.war file, as well as paths to additional servlets or jar files that you want to export.
2. Open a **Terminal** or **Command Prompt** window, then navigate to the directory where the Migration Tool is installed.
3. Issue a command in the **Terminal** or **Command Prompt** windows that indicates the action you want the Migration Tool to perform, as well as the locations to the pentaho-solutions directory, the pentaho.war file, and optionally any additional servlets and jar files that you want to migrate. Also indicate where you want the Migration Tool to place the migration bundle file. Complete these steps based on your operating system.

   The Migration Tool command contains these parameters:

   - **--import:** Indicates this is an import command. This parameter is required if you want to import a migration bundle file.
   - **--migration-bundle-file:** Specifies the directory where you want the migration file to be stored. This parameter is optional. If you do not supply it, the migration bundle file is stored in the same directory where the Migration Tool is installed.
   - **--pentaho-solutions-dir:** Indicates the path to the `pentaho-solutions` directory for the new version of the DI Server. This parameter is required.
   - **--pentaho-war:** Indicates the path to `pentaho.war` file for the new version of the DI Server. This parameter is required.
   - **--extra-paths:** Specifies the path(s) to servlets, JSPs, JDBC drivers, or other additional files that you want migrated to the new version of the DI Server. This parameter is optional.

   a) **Windows:** Enter a command that follows this format in the **Command Prompt** window. The command should be formatted like this.

Command Example:

```
migrator.bat --import --migration-bundle-file=C:\migration.zip --pentaho-
solutions-dir=C:\Users\pentaho\biserver-5.0\biserver-ee\pentaho-solutions --
pentaho-war=C:\Users\pentaho\biserver-5.0\biserver-ee\tomcat\webapps\pentaho
```

b) **Linux:** Enter a command that follows this format in the **Terminal** window. The command should be formatted like this.
Command Example:

```
./migrator.sh --import --migration-bundle-file=/Users/mycompany/migration.zip --
pentaho-solutions-dir=/Users/mycompany/biserver-4.8/biserver-ee/pentaho-solutions
 --pentaho-war=/Users/mycompany/biserver-4.8/biserver-ee/tomcat/webapps/pentaho
```

4. The Migrator Tool runs. The progress of the tool is displayed in the **Command Prompt** or **Terminal** window.

Review Import Logs

To verify that the import occured properly and that data was imported, do these things.

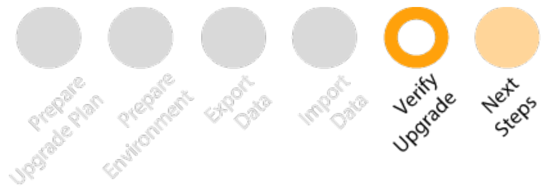
1. Open a **Terminal** or **Command Prompt** window and navigate to the directory where the Migration Tool was installed. The `migrator.log` file should appear. The `migrator.log` file contains a log of the import process.
2. Open the `migrator.log` file and note any failures or errors. If any failures or errors exist, you might need to migrate that data manually.
3. Close the file, then rename the `migrator.log` file to something else.

Manually Migrate Additional Data

These types of items must be manually migrated to the new server.

- JDBC Drivers.
- Kettle plugins. You will need to get new versions of the plugins. If you have written a plugin, you will need to thoroughly test the plugin to ensure that it works with the new version of the system.
- Other web applications installed alongside Pentaho. If you have other web applications that have been installed with Pentaho, you must manually update them.
- Manually copy the server.xml file from TBD to TBD. (TBD - check with Benny)
- Manually copy the context.xml fle from TBD to TBD (TBD - check with Benny)

Verify Upgrade



After you have *imported the data into the new system*, verify that migration and the other parts of the upgrade process occurred properly. Perform these tasks.

- *Start Old Version of the DI Server*
- *Start New Version of the DI Server*
- *Verify Upgrade with Verification Checklists*

Start Old Version of the Server

Start the old version of the server. Wait for the server to start up completely before proceeding to the next task.

Start New Version of the Server

Start the new version of the server. As the server starts, watch the web application server log carefully and note any errors that might have occured. Resolve any port conflict issues as necessary.

Verification Checklists

Verification largely consists of using the checklists to compare the differences between the new and old systems. To make this task easier, it is recommended that you view both systems simultaneously so that you can accurately identify differences.

Overall System

Ensure that the new version of the DI Server is working properly by using the following checklist.

- Did the DI Server start properly? Were you able to log in as admin without issues or problems?
- Did the web application server log have any errors or anomalies?
- If you have configured the DI Server as a service, is it starting and stopping properly?
- Manually stop and restart the DI Server. Did it stop and restart properly?

Users and Roles

To verify that user and role data have been migrated properly, do these things.

1. TBD.
2. TBD.
3. This is the example.

Table 3: User and Roles Verification Checklist

Verification Item	Suggested Verification Approach	Expected Result/Notes
Do the active user accounts in the old system appear in the new system?	Compare the names of users in the old and new systems. • New System: Review the **Manage Users and System Roles** tabs in the **User/Roles** page in the UC.	The active user accounts from the old version of the system should appear in the new system. Note that inactive accounts are not migrated.

Verification Item	Suggested Verification Approach	Expected Result/Notes
	• Old System: Review the **User & Roles** tab in the **Administration** page in PEC.	
Do the roles in the old system appear in the new system?	Compare the roles in the old and new systems. • New System: Review the **Manage Users** and **Manage Roles** tab in the **User/Roles** page in the UC. • Old System: Review the **User & Roles** tab in the **Administration** page in PEC.	The roles from the old version of the system should appear in the new system. If some roles are missing, you might need to manually recreate them.

Shareable Objects

To verify that shareable objects were migrated properly, do these things.

1. Start Spoon.
2. TBD..
3. Verify the items in the checklist.

Table 4: Shareable Object Verification Checklist

Verification Item	How to Check	Expected Result/Notes
For each user, do the report folders, subfolders, and files that were in the old system, appear in the new one? Are the same files in the same folders?	• New System: Review the **Browse Files** page and review the files and folders for each user. • Old System: TBD.	Note that the structure of the report folders, subfolders, and files have changed. Generated content now appears in each user's home directory.
Did the folder permissions from the old version of the system migrate correctly to the new system?	• New system: Review the permissions in the **Share** tab of the **Properties** page. • Old System: Review the permissions in the **Share tab** of the **Properties** page.	Make sure role permissions are set before performing this step. The permissions settings for version 5.0 of the system differ from 4.8. Version 4.8 of the system included these permissions: **Execute**, **Subscribe**, **Create**, **Update**, **Delete**, and **Permissions**. Version 5.0 of the system includes these permissions: **All**, **Read**, **Write**, **Manage Access Control (ACL) Management.** Mappings between 4.8 and 5.0 are: • In version 5.0, the **All** permission is assigned to folders that were assigned **Execute**, **Subscribe**, **Create**, **Update**, **Permissions**, AND **Delete** accesses in version 4.8 of the system. • In version 5.0, the **Read** permission is assigned to folders that were assigned **Execute** and/or **Subscribe** permissions in version 4.8 of the system.

Verification Item	How to Check	Expected Result/Notes
		• In version 5.0, the **Write** permission is assigned to folders that were assigned **Create**, **Update**, and/or **Delete** permissions in version 4.8 of the system. • In version 5.0, the **Manage Access Control (ACL) Management** permission is assigned to folders that, in version 4.8 of the system, **Permission** access.
Did the folder permissions from the old version of the system migrate correctly to the new system?	• New system: Review the permissions in the **Share** tab of the **Properties** page. • Old System: Review the permissions in the **Share** tab of the **Properties** page.	Make sure role permissions are set before performing this step. The permissions settings for version 5.0 of the system differ from 4.8. Version 4.8 of the system included these permissions: **Execute**, **Subscribe**, **Create**, **Update**, **Delete**, and **Permissions**. Version 5.0 of the system includes these permissions: **All**, **Read**, **Write**, **Manage Access Control (ACL) Management.** Mappings between 4.8 and 5.0 are: • In version 5.0, the **All** permission is assigned to folders that were assigned **Execute**, **Subscribe**, **Create**, **Update**, **Permissions**, AND **Delete** accesses in version 4.8 of the system. • In version 5.0, the **Read** permission is assigned to folders that were assigned **Execute** and/ or **Subscribe** permissions in version 4.8 of the system. • In version 5.0, the **Write** permission is assigned to folders that were assigned **Create**, **Update**, and/or **Delete** permissions in version 4.8 of the system. • In version 5.0, the **Manage Access Control (ACL) Management** permission is assigned to folders that, in version 4.8 of the system, **Permission** access.
Do users and roles have access to files and reports that they should *not* have access to? Make sure that you check the permissions for both Mondrian and metadata model-based reports.	• New System: Create test user accounts for each role in your system. Log into the UC using the test user accounts and attempt to access files that users assigned that role should or should not have access to.	Ensure that users and roles do not have access to restricted files.

Verification Item	How to Check	Expected Result/Notes
	Record results and adjust permissions (role or report) as necessary. Compare with the roles on the old system.	
Did the schedules for the reports that were in the old system migrate properly? See expected results/ notes for public vs. private schedule changes.	- New System: Review the **Schedules** page in the UC. - Old System: Review the **Scheduler** tab in the **Administration** page in PEC.	There are no longer public and private schedules. Instead, there are only one set of schedules. You can assign parameters to a schedule. If there is one public schedule that had many subscribers, each subscriber's private schedule is now converted to a schedule. So, if in the old system you had a public schedule that had four subscribers, four schedules will appear in the new system. If your schedule runs using a cron job, you might want to edit it so that it is in a more easily readable format. Schedules should be paused on the new system. If a public schedule had no subscribers in the old system, it is not migrated.
Did the generated content (e.g. the PDFs of previous report runs) on the old system migrate correctly?	- New System: These folders appear in each users home directory and might need to be tested by the users themselves.	If a user was deleted or is inactive, the generated content is not migrated to the new system.
Are the data sources that were in the old system migrate properly to the new system?	- New System: Review the **Data Sources** page in the UC. - Old System: Review the **Manage Data Sources** page in the UC.	If this information did not migrate properly, it might be necessary to manually add the data sources.
Did the JDBC database connections for the old system migrate properly to the new system? Is the connection information for each correct?	- New System: Edit a data source from the **Data Sources** page, then click General in the **Database Connection** page. Click the **Test** button in that page to ensure that the connection is still valid. - Old System: Review the **Manage Data Sources** page in the UC (TBD).	If this information did not migrate properly, it might be necessary to manually edit them.
Are the advanced settings configured properly for each database connection?	- New System: Edit a data source from the **Data Sources** page, then click the different tabs in the page. - Old System: Review the **Manage Data Sources** page in the UC (TBD)	If this information did not migrate properly, it might be necessary to manually edit them.
Open and inspect interactive reports. Ensure that they are displaying properly.	- New System: Retrieve reports from **Browse Files** page in the UC. - Old System: Retrieve reports from the **Browse** page in the UC.	If a report does not display properly, and the SQL command contains several joins, ensure the join order has been specified for that report. See the *Troubleshooting* in the

Verification Item	How to Check	Expected Result/Notes
		Work with Relational Data Models section for details.
Open and inspect dashboard reports. Verify that each part of the dashboard displays properly.	• New System: Retrieve reports from **Browse Files** page in the UC. • Old System: Retrieve reports from the **Browse** page in the UC.	If a report does not display properly, and the SQL command contains several joins, ensure the join order has been specified for that report. See the *Troubleshooting* in the Work with Relational Data Models section for details.
Ensure that drill linking (connections between reports) is working properly.	• New System: Retrieve reports from **Browse Files** page in the UC. • Old System: Retrieve reports from the **Browse** page in the UC.	

System Configuration Settings

To ensure that system configuration settings have migrated properly, use the following checklist.

Table 5: System Configuration Settings Verification Checklist

Verification Item	How to Check	Expected Result/Notes
Verify that the server base URL, fully qualified server URL, and the locale language in the new server match what they were in the old server.	• New System: Check the `web.xml` file. • Old System: In PEC, click the **Web Settings** tab in the **Configuration** page.	If you have customized the old system with servlets, filters, context parameters or in other ways, you will need to copy these things into the new `web.xml` file.
Build a new analyzer report and verify that the custom chart properties appear.	• New System: TBD. (They need to build a new analyzer report)	Custom chart properties should appear.
Verify that the email server settings in the new system match the settings in the old one.	• New System: Review the **Mail Server** page. • Old System: In PEC, click the **BI Components** tab in the **Configuration** page.	The email server settings should match.
Verify that custom LDAP settings in the new server match those for the old server.	• New System: View the **Authentication** page in the UC. • Old System: View the **LDAP** tab in the **Utilities** page.	The LDAP settings should match.
Verify that the custom log4j configuration settings in the old system match those in the old one.	• New system: View the file in this file path: `webapps\pentaho\WEB-INF\classes\log4j.xml` • Old System: TBD.	The log4j settings should match.
Review the `settings.xml` files in each plugin on the new server and compare to the old one.	Use a file difference tool to compare the files in the old and new systems.	Review the differences between the two files. If customized changes were made, they should have migrated properly. If not, you will need to manually migrate them.

Verification Item	How to Check	Expected Result/Notes
Compare the settings in these files: `applicationContext-security-ldap.properties`, `applicationContext-spring-security-hibernate.properties`, `pentaho.xml`, `mondrian.properties`, `settings.xml` for various plugins, `analyzer.properties`, `email_config.xml`. Also the following tomcat files: `web.xml, context.xml, classic-engine.properties, logging.properties, log4j.xml`	Use a file difference tool to compare the files in the old and new systems.	Review the differences between the two files. If customized changes were made, they should have migrated properly. If not, you will need to manually migrate them. If you update the `context.xml` file, you will need to delete the `pentaho.xml` that is in the `tomcat\conf\Catalina` directory before you start the DI Server again for the changes to take effect.

Users Settings

To verify that user settings were migrated properly, complete these steps.

1. Have each user log into the User Console.

2. Have each user verify the items in the checklist.

Table 6: User Settings Verification Checklist

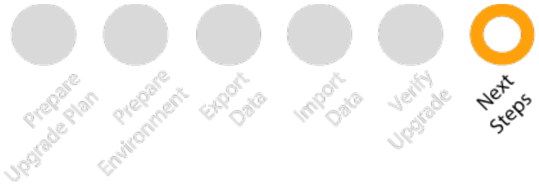
Verification Item	How to Check	Expected Result/Notes
Verify that your recent files list on the new system matches what was in the old one.	• New System: Log into the UC, then view **File > Recent** • Old System: Log into the UC, then view **File > Recent**	If this did not migrate properly, you will need to add this manually.
Verify that your favorites list on the new system matches what was in the old one.	• New System: Log into the UC, then view **File > Favorites** • Old System: Log into Mobile, then TBD.	This appears in Mobile only. If this did not migrate properly, the user will need to add this manually.
Verify that your “show descriptions for tooltips” setting on the new system matches what was in the old one.	• New System: Log into the UC, then view **View > Use Descriptions** for Tooltips • Old System: TBD.	If this did not migrate properly, you will need to add this manually.

Manual Configuration Settings

To ensure that manual configuration settings have been properly set, use the following checklist.

TBD.

Next Steps



After you have *verified the upgrade* and upgraded Spoon, perform post-migration tasks. These include completing the following tasks.

- Configure the DI Server.
- To learn more about how to administer the DI Server, read Administer DI Server.
- Deprovision the old server and provision the new one in accordance with your organization's procedures.
- Review the What’s New in DI if you have not done so already.
- Review Getting Started with Pentaho to learn how to use Spoon.