



BEA AquaLogic® Enterprise Repository

Upgrade Guide

Version 3.0 RP1
Revised: February, 2008



Table of Contents

How to Upgrade from ALER 3.0 to ALER 3.0 RP1



- [ALER 3.0 to 3.0 RP1 Upgrade Process Overview and Requirements](#)
- [ALER 3.0 to 3.0 RP1 Tomcat Application Upgrade](#)
- [ALER 3.0 to 3.0 RP1 WebLogic Server Application Upgrade](#)
- [ALER 3.0 to 3.0 RP1 WebSphere Application Upgrade](#)
- [Post-Installation Tasks](#)

How to Upgrade from ALER 2.6 to ALER 3.x

- [ALER 2.6 to 3.x Upgrade Process Overview and Requirements](#)
- [Database Upgrade](#)
 - [Create Upgrade Working Directories](#)
 - [Extract the Upgrade Package ZIP File](#)
 - [Backup Current Database](#)
 - [Create a sql_temp Directory](#)
 - [Copy Files](#)
 - [Modify SQL Scripts](#)
 - [Migrate the Data](#)
 - [Resolving Migration Encoding Issues](#)
- [Create an ALER 3.0 Database For Use With the ALER Installer](#)
 - [Create Oracle Database](#)
 - [Create SQL Server Database](#)
 - [Create UDB/DB2 Database](#)
- [Tomcat Application Upgrade](#)
 - [Create an aler_temp Directory](#)
 - [Back Up the Current Installation](#)
 - [Run the ALER Product Installer](#)
 - [Restore Files](#)
 - [Start the Application Server](#)
 - [!\[\]\(aca6fcc8bd95e8255b9ea1b1d08ef300_img.jpg\) \[Tomcat Post-Installation Instructions\]\(#\)](#)

- **WebLogic Server Application Upgrade**
 - Create an aler_temp Directory
 - Back Up the Current Installation
 - Remove the Current Installation Files
 - Run the ALER Product Installer
 - ALER and Workflow Database Installation
 - Restore Files
 - Start the Application Server
 -  [WebLogic Server Post-Installation Instructions](#)
- **WebSphere Application Upgrade**
 - Create an aler_temp Directory
 - Back Up the Current Installation
 - Remove the Current Installation Files
 - Run the ALER Product Installer
 - Install New Application Files
 - Restore Files
 - Configure Settings
 - Add a Shared Library to the Application Server
 - Create a BEA_HOME WebSphere Variable
 -  [WebSphere Post-Installation Instructions](#)
- **Post-Installation Tasks**

Post-Installation Tasks



- **Migrating Custom Reports**
- **Remove Previous Version of Eclipse Plug-in on Client Workstations**
-  [Install WorkSpace Studio on Client Workstations](#)
-  [Import the AquaLogic Solution Package for Assembly Models](#)
- **Verify the Value of the Assets-In-Progress Setting**
- **Using the Diagnostics Testing Tool**
 - Navigating to the Diagnostic Page
 - Links
 - Checking Product Version Information
 - Test Servlet Functionality
 - Test Required Libraries
 - Test Database Connectivity
 - List System Paths
 - Run Pre-compile Servlet
 - Other Features
 - Test the Web Server Installation
- **Using an X Windows Server with ALER Reporting Engine**
 - Granting Access to an X Windows Server
 - Setting DISPLAY Variable
 - Other Notes
 - X Windows Alternatives

ALER 3.0 to 3.0 RP1 Upgrade Process Overview and Requirements

This section provides an overview of the upgrade process tasks and requirements when upgrading from version 3.0 to 3.0 RP1. Use the following links for additional instructions specific to your application server

- [Tomcat Application Upgrade](#)
- [WebLogic Server Application Upgrade](#)
- [WebSphere Application Upgrade](#)


Upgrade Process

1. [Backup the existing ALER database](#)
2. Backup existing ALER application deployment:
3.  [Verify required platforms and version information \(including JDBC Driver version support\)](#)
4. Uninstall ALER 3.0
5.  [Run the ALER 3.0 RP1 product installer](#)
6. Restore the ALER backup files from the previous ALER deployment to the newly installed ALER 3.0 RP1 version, including:
 - The ALER property files
 - If you are using LDAP: confirm **Enable LDAP Login** is set to **True** and verify the **LDAP Configuration**. If you are using SSO: confirm **Enable Advanced Container Login** is set to **True** and verify the **Advanced Container Configuration** settings. If your LDAP or SSO configurations are not set correctly, users will not be able to authenticate with the ALER application.
 - Any custom files, such as custom home pages, images, XSLT templates, upload files, and any other customer-related files
7. Start your application server.
8. [Verify the application deployment using ALER diagnostics](#)

Tomcat Application Upgrade

These instructions enable you to upgrade from ALER 3.0 to ALER 3.0 RP1 on Tomcat using the BEA Products installation program.

Prerequisites

- Please reference the  [Supported Configurations](#) list for the most recent information regarding your application server version.
- If the installed application server version is no longer supported by ALER, or the application server has been upgraded to a newer version, follow the application upgrade instructions for the application server side, and then continue following the database upgrade instructions within this document.
- The directory where Tomcat is installed on your system is referenced in these instructions as `CATALINA_HOME`.
- The web application context that is being upgraded is referenced in these instructions as `CATALINA_HOME/webapps/aler`.
- **Backup the existing ALER database**

Create an `aler_temp` Directory

On the Tomcat server, create a directory named `aler_temp`. This directory will contain the backup copies of the current ALER application. This will also serve as a temporary location for properties files, JDBC drivers, and any customized ALER items (`home.jsp` for example) that are required for the new application.

Back Up the Current ALER Installation

Navigate to the Tomcat installation location `CATALINA_HOME/webapps` and copy the existing `aler` and `aler-web` application directories into the `aler_temp` directory.

Uninstall ALER 3.0.

Use the ALER 3.0 Installer to remove the previous release. See  [Uninstalling the Software](#)

Run the BEA Products Installation Program

1. When prompted for the **Tomcat (Catalina) Home** directory, supply the directory where the previous deployment had been installed.
2. When prompted for the **Repository application name**, use the same application name that the previous deployment had used.
3. When prompted for the **Port number**, use the same port number that the previous deployment had used.
4. If you encounter a “One or more installation directories are not empty. Proceed with installation?” message, select YES.

The installer will overwrite the existing deployment with the new files, thus preserving any additional custom files (such as an upload repository directory, custom home pages, etc.).

Post-Installation Instructions

See  [Stand-alone Tomcat Installations](#)

Restore the ALER Backup Files

1. Navigate to the `aler_temp` directory.
2. Copy the `aler_temp/aler/WEB-INF/classes/*.xml` files to the `CATALINA_HOME/webapps/aler/WEB-INF/classes` directory. This action overwrites the XML files that are a part of the new install.
3. Copy the `aler_temp/aler/WEB-INF/classes/*.properties` files to the `CATALINA_HOME/webapps/aler/WEB-INF/classes` directory. This action overwrites the XML files that are a part of the new install.
4. Copy the `aler_temp/aler/custom/home.jsp` (and/or all of the custom home pages) to the `CATALINA_HOME/webapps/aler/custom` directory. This overwrites the existing `home.jsp` file that is part of the new install. In addition, copy any custom images that were included in the custom home page(s) to the appropriate directory.
5. If using any custom Type icons, copy the files from the `aler_temp/aler-web/images/tree/enterprise/asset` directory to the `CATALINA_HOME/webapps/aler-web/images/tree/enterprise/asset` directory.
6. If custom reports were created, see the [Migrating Custom Reports](#) section.
7. Copy/Move the Upload Repository directory (normally found in `aler_temp/aler/custom/upload`) into the new `CATALINA_HOME/webapps/aler/custom` installation directory.

Start Tomcat

Important! If you are using LDAP: confirm **Enable LDAP Login** is set to **True** and verify the **LDAP**

Configuration. If you are using SSO: confirm **Enable Advanced Container Login** is set to **True** and verify the **Advanced Container Configuration**. If your LDAP or SSO configurations are not set correctly, users will not be able to authenticate with the ALER application.

1. Navigate to the `CATALINA_HOME/work` directory and delete the cache for the ALER 3.0 application.
2. Start the Tomcat application server.

Test the Application

ALER 3.0 RP1 is now deployed. Testing the installation through the use of the ALER Diagnostics page is recommended. Recommended tests are:

- Product Version Information
- Test Required Libraries
- Test Database

WebLogic Server Application Upgrade

These instructions enable you to upgrade from ALER 3.0 to ALER 3.0 RP1 on WebLogic Server using the BEA Products installation program.

Prerequisites

- ALER 3.0 RP1 requires WebLogic Server 9.2 MP2. If the currently installed ALER application is not running on the required version, upgrade the application server and the existing ALER domain appropriately.
- It is assumed that the ALER domain is installed within the default location of `BEA_HOME/user_projects/applications/aler`. If the ALER domain is installed in a different location, then make the appropriate adjustments to these instructions.
- **Backup the existing ALER database**

Create ALER Temp Directories

- On WebLogic Server, create a directory named `aler_upgrade`. This directory is used to contain the ALER upgrade files.
- On WebLogic Server, create a directory named `aler_temp`. This directory is used to backup the current ALER application. This location is used as a temporary location for properties files, JDBC drivers, and any customized ALER items (`home.jsp`, for example) that are necessary for the new application.

Back Up the Current ALER Installation

1. Log into the WebLogic Server Administration Console and select the ALER deployment.
2. Choose the option to `Lock & Edit`
3. Choose the option to stop the application and confirm the choice if required.
4. Select the ALER application and choose the option to uninstall the ALER deployment. This operation will remove the deployment from the domain's `config.xml` file without removing any of the files within the deployment. The ALER Product Installer application will create a new domain for the

ALER 3.0 installation. Therefore the existing domain will no longer be used.

5. Navigate to the `BEA_HOME/user_projects/applications/aler` directory and copy the `aler-app` and `aler-web` directories to the `aler_temp` directory.

Remove Current ALER Installation Files

Caution! Perform this step only after completing backup of the directory, as described in [Back Up the Current ALER Installation](#).

Remove the ALER application `BEA_HOME/user_projects/applications/aler/aler-app`, `BEA_HOME/user_projects/applications/aler/aler-app` and `BEA_HOME/user_projects/applications/aler/META-INF` directories.

Uninstall ALER 3.0.

Use the ALER 3.0 Installer to remove the previous release. See  [Uninstalling the Software](#)

Run the BEA Products Installation Program

1. When prompted for the **BEA Home** directory, supply the directory where the WebLogic Server 9.2 MP2 is installed.
2. When prompted for the **Domain name** select the same domain name that had been deployed previously.
3. When prompted for the **Fully qualified server name**, supply the fully qualified host name of the server that this domain is being installed on.
4. When prompted for the **Repository application name**, use the same application name that the previous deployment had used.
5. When prompted for the **Port number**, use the same port number that the previous deployment had used.

ALER and ALBPM Workflow Database Installation

1. If you plan to install the Workflow option as a part of this upgrade, then when prompted to configure the database, choose **Yes**. This will be a required process for the Workflow deployment. Otherwise, you may choose the **No** option to bypass this and the following step.
2. A second ALER database will be necessary to allow the BEA Products installer to pass to the Workflow database and application configuration stage. Once this process is completed, this second database can be dropped without impact on the upgraded deployment.

Post-Installation Instructions

See  [WebLogic Server Installations](#).

Restore the ALER Backup Files

1. Navigate to the `aler_temp` directory.
2. Copy the `aler_temp/aler-app/WEB-INF/classes/*.xml` files to the `BEA_HOME/user_projects/domains/<domain>/aler-app/WEB-INF/classes` directory. This action overwrites the XML files that are a part of the new install.
3. Copy the `aler_temp/aler-app/WEB-INF/classes/*.properties` files to the `BEA_HOME/user_projects/domains/<domain>/aler-app/WEB-INF/classes` directory. This action overwrites the XML files that are a part of the new install.
4. Copy the `aler_temp/aler/custom/home.jsp` (and/or all of the custom home pages) to the `BEA_HOME/user_projects/domains/<domain>/aler-app/custom` directory. This overwrites the existing `home.jsp` file that is part of the new install. In addition, copy any custom images that were included in the custom home page(s) to the appropriate directory.
5. If using any custom Type icons, copy the files from the `aler_temp/aler-web/images/tree/enterprise/asset` directory to the `BEA_HOME/user_projects/domains/<domain>/aler-web/images/tree/enterprise/asset` directory.
6. If custom reports were created, see the [Migrating Custom Reports](#) section.
7. Copy/Move the Upload Repository directory (normally found in `aler_temp/aler/custom/upload`) into the new `BEA_HOME/user_projects/domains/<domain>/aler-app/custom` installation directory.

Start WebLogic Server

Important! If you are using LDAP: confirm **Enable LDAP Login** is set to **True** and verify the **LDAP Configuration**. If you are using SSO: confirm **Enable Advanced Container Login** is set to **True** and verify the **Advanced Container Configuration**. If your LDAP or SSO configurations are not set correctly, users will not be able to authenticate with the ALER application.

1. Delete the application server cache located within `ALER_DOMAIN/servers/AdminServer/tmp/WL_user/aler` directory. This directory contains compiled JSP pages that pertain to the old application deployment, which are no longer valid.
2. Start the application through the WebLogic Server Administration Console.

Test the Application

ALER 3.0 RP1 is now deployed. Testing the installation through the use of the ALER Diagnostics page is recommended. Recommended tests are:

- Product Version Information
- Test Required Libraries
- Test Database

WebSphere Application Upgrade

These instructions enable you to upgrade from ALER 3.0 to ALER 3.0 RP1 on WebSphere using the BEA Products installation program.

Prerequisites

- ALER 3.0 RP1 requires WebSphere version 6.1.0.5. If the currently installed ALER application is not running on the required version, upgrade the application server as appropriate.
- If the application server version on which the currently installed version of Flashline/ALER is running is no longer supported, or if the application server has been upgraded to a new version, follow the application instructions for the application server side, and then continue following the database upgrade instructions in this section.
- The current deployment directory containing the ALER application is referred to as `aler_current` (this directory will normally be located in the following directory: `WAS_HOME/AppServer/profiles/AppSrv01/installedApps/localhostNode01Cell/installedApps/aler.ear`).
- Once the upgrade process is completed, the ALER property, XML and JDBC driver files will be located within a new `BEA_HOME/repository30/websphere` directory.
- **Backup the existing ALER database**

Create an `aler_temp` Directory

On the WebSphere server, create a directory named `aler_temp`. This directory will contain backups of the current ALER application properties files, JDBC drivers, and any customized AquaLogic Enterprise Repository items (`home.jsp` for example) that are necessary for the new application.

Back Up the Current ALER Installation

1. Copy the contents of the current `ALER_HOME` shared library directory contents to the `aler_temp` directory. The `aler_temp` directory should then contain all of the original ALER properties files, XML files, and your current JDBC driver JAR files.
2. If you have a customized ALER home page, navigate to the `aler_current/aler-app.war/custom` directory and copy the `home.jsp` file to the `aler_temp` directory. If you have used the Role or

Department based home pages, copy those files from the directory that is indicated within the *aler_current/aler-app.war/WEB-INF/classes/homepages.xml* file.

3. Copy any custom images that are used in the custom home page file(s) to the *aler_temp* directory.
4. If you have any custom Asset Type icons, navigate to the *aler_current/aler-web.war/images/tree/enterprise/asset* directory and copy the images to the *aler_temp* directory.
5. Optionally, copy all XSLT templates from the PDF Printing web-application to the *aler_temp* directory. This web application is bundled directly into the core of the ALER product.
6. Copy any additional files or directories that were included or modified from the base ALER installation into the *aler_temp* directory.

Remove the Current ALER Installation Files

1. Start the WebSphere 6.1.x Administrative Console.
2. Expand the Applications -> Enterprise Applications folder and select the **ALER** check box.
3. Click **Stop**.
4. Select the **ALER** check Box.
5. Click **Uninstall**.
6. Repeat the above steps to remove the PDF Printing web-application (if present). This feature is now bundled within the ALER base application.
7. Save the changes to the Master Configuration when prompted.

Uninstall ALER 3.0.

Use the ALER 3.0 Installer to remove the previous release. See  [Uninstalling the Software](#)

Run the BEA Products Installation Program

1. When prompted for the **Repository application name**, use the same application name that the previous deployment had used.
2. When prompted for the **Port number**, use the same port number that the previous deployment had used.

The BEA Products installer will generate a deployable EAR file, thus allowing you to quickly and easily install the updated version of the application directly from the application server's file system. It will also create and deploy the ALER Workflow application components appropriately.

Install New Application Files

1. Start the WebSphere Administrative Console.
2. Expand the Applications -> Enterprise Applications folder and click **Install**.
3. Choose the option to select a **Remote Filesystem** (this option is relative to the application server), browse to the `BEA_HOME/repository30/websphere` directory, and select the `aler.ear` file.
4. Select the **Prompt me only when additional information is required** option and click **Next** to continue.
5. On the Select Installation Options page, click **Next** to continue after making the appropriate changes.
6. On the Map modules to servers page, click **Next** to continue after making the appropriate changes.
7. On the Map virtual hosts for Web modules page, click **Next** after making the appropriate changes.
8. On the Deployment summary page, click **Finish** to deploy the EAR file.

Note: This process may take several minutes.

9. After the installation is finished, the **Application installed successfully** message is displayed.
10. Click **Save to Master Configuration** when prompted.

Restore the Backup ALER Files

1. If you have a customized ALER home page(s), copy `home.jsp` to the `aler_current/aler-app.war/custom` directory. This operation will overwrite the `home.jsp` file that is part of the new install.
2. Copy any custom images that may be part of the `home.jsp` file to the appropriate directory.
3. If using any custom Type icons, copy the files to the `aler_current/aler-web.war/images/tree/enterprise/asset` directory.
4. If custom reports were created, see the [Migrating Custom Reports](#) section.
5. If additional files were included or modified in the original ALER install, consult BEA Technical

Support for assistance with inclusion of these additions into ALER 3.0.

6. Copy all previous property files from the *aler_temp* to the *BEA_HOME/repository30/websphere* directory.
7. Copy all previous XML files from the *aler_temp* to the *BEA_HOME/repository30/websphere* directory.
8. Copy the currently supported JDBC Driver files into the *BEA_HOME/repository30/websphere* directory.

Start WebSphere

Important! If you are using LDAP: confirm **Enable LDAP Login** is set to **True** and verify the **LDAP Configuration**. If you are using SSO: confirm **Enable Advanced Container Login** is set to **True** and verify the **Advanced Container Configuration**. If your LDAP or SSO configurations are not set correctly, users will not be able to authenticate with the ALER application.

1. Navigate to the *WebSphere_HOME/AppServer/temp/* directory and delete the cache for the ALER application.
2. Restart WebSphere.

Test the Application

ALER 3.0 RP1 is now deployed. Testing the installation through the use of the ALER Diagnostics page is recommended. Recommended tests are:




- Product Version Information
- Test Required Libraries
- Test Database

ALER 2.6 to 3.x Upgrade Process Overview and Requirements

This section provides an overview of the upgrade process tasks and requirements when upgrading from ALER 2.6 to v3.X. Use the following links for additional instructions specific to your application server

- [Tomcat](#)
- [WebLogic Server](#)
- [WebSphere](#)

Upgrade Process Overview

1. **Backup the existing ALER database**
2. Backup existing ALER application deployment:
3. Remove previous ALER `pdfreport` Web application (this feature is delivered as a part of the ALER 3.0 installation)
4.  [Verify required platforms and version information \(including JDBC Driver version support\)](#)
5. **Create a new database (SQL Server or UDB) or user (Oracle) for use with the ALER installer application**
6. Download the ALER 3.0 Upgrade Pack ZIP file from the  [Customer Access Portal](#)
7. As part of the migration process, copy the following files from the ALER 3.0 Upgrade Pack to temporary migration directories:
 - Copy the SQL files from the Upgrade Pack ZIP file during the upgrade process
 - Copy the properties files from the current ALER deployment (`database.properties` and `settings.properties`) to the migration `/classes` directory
 - Copy the JDBC driver file(s) to the migration `/lib` directory
8. **Perform the ALER 3.0 database upgrade**
9.  [Run the ALER 3.0 product installer](#)
10. Configure the 3.0 ALER deployment by copying the following files from the previous ALER deployment to the newly installed ALER 3.0 version, including:
 - The ALER property files
 - If you are using LDAP: confirm **Enable LDAP Login** is set to **True** and verify the **LDAP Configuration**. If you are using SSO: confirm **Enable Advanced Container Login** is set to **True** and verify the **Advanced Container Configuration**. If your LDAP or SSO configurations are not set correctly, users will not be able to authenticate with the ALER application.
 - Any custom files, such as custom home pages, images, XSLT templates, upload files, and any other customer-related files
 - The supported JDBC drivers
11. Start your application server.
12. **Verify the application deployment using ALER diagnostics**

Database Upgrade

Important! If the database server version in use is no longer supported for use with ALER, or if the database server has been upgraded to a new version, migrate the database to the new database server and follow the instructions provided in both the database and application upgrade portions of this document.

Prerequisites for Database Upgrade

The database upgrade process can be performed on workstations that conform to the following requirements

- Network connectivity to the ALER database server.
- A JRE or JDK 1.4.2 or 1.5.0 installed to complete the database upgrade process.
- An SQL client tool to execute SQL scripts against the ALER database.


Step 1: Create Upgrade Working Directories

Create the following ALER upgrade and backup working directories

1. Create a directory to contain the upgrade files called *aler_upgrade*
2. Create a directory to contain a backup of the current ALER application files called *aler_backup*

Step 2: Extract the Upgrade Package ZIP File

Before beginning the database upgrade, you must download the 3.0 Upgrade Package from the ALER Customer Access Portal (CAP).

1. Download the Upgrade Database package from  CAP
2. Unzip the ALER30-CORE-DBUpgradeOnly.zip file into the *aler_upgrade* directory created above. The *aler_upgrade* directory should then contain the following:
 - database-scripts/
 - database-lib/
 - migration/
 - oss_licenses/

Step 3: Backup the Current Database

Create a complete backup of the ALER database, including all tables owned by the *db.username* property located in the *database.properties* file. The *database.properties* file can be found in the *aler/WEB-INF/classes* directory on Tomcat application servers and within WebLogic Server, this file can be found in the *aler-app/WEB-INF/classes* directory. For WebSphere application servers this file is normally stored externally to the application EAR file and is referenced using a shared library.

Step 4: Create a sql_temp Directory

On the database server, create a directory named *sql_temp*.

Step 5: Copy Files

1. Navigate to the *aler_upgrade* directory.
2. Locate the file that is relevant to your database:
 - o SQL Server: *database-scripts/mssql/upgrade26-30/**
 - o IBM UDB: *database-scripts/udb/upgrade26-30/**
 - o Oracle: *database-scripts/oracle/upgrade26-30/**
3. Copy the appropriate upgrade file to the *sql_temp* directory.
4. Navigate to the *aler_upgrade* directory and copy the *migration/* directory to the *sql_temp* directory.
5. Navigate to the *aler/WEB-INF/classes* directory on the application server and copy the *database.properties* file to the *sql_temp/migration/classes* subdirectory.

Note: If you are planning on supporting Unicode in your Repository install, add the property "db.unicode=true".

6. Navigate to the *aler/WEB-INF/lib* directory on the application server and copy the JDBC driver file(s) to the *sql_temp/migration/lib* subdirectory.

Note: ALER 3.0 only supports the use of the IBM UDB Type 4 JDBC Drivers. The important differences in the IBM UDB Type 4 JDBC Drivers are listed below:

- o Type 4 Drivers are packaged in *db2jcc.jar*, *db2jcc_javax.jar* and *db2jcc_license_cu.jar*
- o Type 4 Driver class name: *com.ibm.db2.jcc.DB2Driver*
- o Type 4 Driver connection class: *com.flashline.db.ext.IBMType4UDBConnection*
- o An example ALER *database.properties* file utilizing the Type 4 Drivers is listed below
 - *db.url=jdbc:db2://db.example.com:50001/aler*
 - *db.username=aler*
 - *db.password=aler_password*
 - *db.driver=com.ibm.db2.jcc.DB2Driver*
 - *db.connectionclass=com.flashline.db.ext.IBMType4UDBConnection*
 - *db.usepool=true*
 - *db.unicode=true*
- o Copy the Type 4 driver JARs (*db2jcc.jar*, *db2jcc_javax.jar* and *db2jcc_license_cu.jar*) files into the application server's *aler/WEB-INF/lib* directory for Tomcat servers. For WebLogic

you would copy these files into the `aler-app/WEB-INF/lib` directory. With WebSphere servers, these files would be copied into the shared library directory that the ALER application deployment references.

- Modify the **aler/WEB-INF/classes/database.properties** file to associate the application with the new drivers just installed as outlined in the examples above.
- Use the ALER Diagnostic Installation Tests page **Test Database** link to verify which JDBC Driver and version number being used to connect to the ALER database.
- Use the ALER Diagnostic Installation Tests page **Snoop Servlet** link to verify the container's CLASSPATH settings. Verify that the appropriate JDBC JAR files are being referenced and that the previous versions are not included within this CLASSPATH listing. You should also verify that the order in which the database drivers appear within the CLASSPATH also matters. The drivers should appear prior to any other references that are either external to the ALER application, or external to the Shared Library directory definition within WebSphere environments.

Step 6: Modify the SQL Scripts

The `sql_temp` directory contains the following scripts:

- *10DropForeignKeys.sql*
- *20PreDataSchemaChanges.sql*
- *30PerformDataMigration.sql*
- *40PostDataSchemaMigration.sql*
- *50Deprecation.sql*

1. Modify the upgrade scripts by replacing `SCHEMA_OWNER`, `DATA_TABLESPACE`, `INDEX_TABLESPACE` with the proper values.
2. SQL server tablespaces actually refer to filegroups. Replace `DATA_TABLESPACE` with `PRIMARY` **and** replace `INDEX_TABLESPACE` with `INDEX`. Assuming that those are the values you have assigned to your respective MDF files upon initial setup of Flashline/ALER SQL server database.

Step 7: Migrate the Data

1. Open a SQL tool such as SQLPlus, Toad, db2client or SQL Server Studio Management Console.
2. Run the following SQL files from the `sql_temp` directory:
 - *10DropForeignKeys.sql*
 - *20PreDataSchemaChanges.sql*
 - *30PerformDataMigration.sql*
3. Navigate to the `migration/` subdirectory within the `sql_temp` directory.
4. Select one of the `env-` files based on the database selected for upgrade and rename it `env.bat` or `env.sh` (use `.bat` extension for Windows and `.sh` extension for UNIX/Linux).

5. Copy the settings.properties from your existing Flashline/ALER application's WEB-INF/classes (or shared library in the case of WebSphere) directory into the `sql_temp/migration/classes/` directory prior to running the migrate batch file. This property file will no longer be needed within ALER 3.0 or higher applications, but is *required* for the upgrade process.
6. Using Java 1.4.2 or 1.5.0 installed, run the `migrate.bat` or `migrate.sh` script.

This script migrates the data in the database to the new database schema structure defined by the SQL scripts run previously. Depending on the amount of data in the database and the speed of the computer this may take several minutes.

7. Once the Java-based data migration is completed successfully, run the following SQL scripts in the `sql_temp` directory to complete the database upgrade process:
 - o `40PostDataSchemaMigration.sql`
 - o `50Deprecation.sql`

Note: When running the java process during the upgrade installation, larger data sets may run into JVM memory constraints. If *out of memory* errors occur while running the `migrate.sh` or `migrate.bat` process, increase the memory as follows:


1. Open the `r4.bat/r4.sh` file.
2. Increase the memory by increasing the `java -mx192m` setting (for example:
`java -mx1024m`).
3. Save the changes.
4. Re-run the `migrate.sh` or `migrate.bat` process.

Step 8: Resolve Migration Encoding Issues

In ALER 2.6 or higher, all BLOB data is stored internally in UTF-8 format. In prior versions, text stored in BLOBs was encoded using the default encoding of the JVM, normally ISO-8859-1.

1. Edit the `env.bat` or `env.sh` pertinent to the database platform being used and uncomment the `MIGRATE_OPTS=-Daler.encoding=UTF-8` line.
2. Modify the value "UTF-8" to match the encoding used when writing the BLOB content.

This will likely be the default encoding of the JVM that was being used when the BLOBs were written, and can be determined by inspecting the documentation for the JVM. Also, check with your system administrator to determine if the default encoding has been overridden within the database.

For more details regarding UTF-8 support within your database definition, refer to the  ["Database Configuration for UTF-8/Unicode Character Encoding"](#) instructions.

3. Start the migration.

The database upgrade is now complete.

Create a Database For Use by the BEA Products Installation

The BEA Products installation program for ALER can be used to upgrade the existing ALER web application, as well as to deploy the ALER Advanced Registration Flow tool and ALBPM workflow databases. The ALER and ALBPM workflow database configuration processes are run in tandem; therefore, a new database to be used by the ALER database configuration process is required to allow the product installer to create and deploy the ALBPM databases and workflows.

This ALER database/user created during the upgrade process are temporary, and so should be dropped after the upgrade install is completed.

- [Create Oracle Database](#)
- [Create SQL Server Database](#)
- [Create UDB/DB2 Database](#)

Oracle Database Installation

Appropriate administrative privileges are necessary to complete this process. These instructions assume that Oracle SQL *Plus is used to run scripts.

Prerequisites

1. A login with database administrator privileges
2. JDBC drivers for the database server - *ojdbc14.jar* (driver version 10.2.0.2)
3. JDK 1.4.2 or higher on the current workstation
4. SQL *Plus Client tools

Create Tablespaces

1. Log into Oracle (system or DBA privileges required).
2. Create a data tablespace named *aler_data* with at least 300M of available space.
3. Create a second data tablespace named *_aler_lob_* with at least 300M of available space.
4. Create an index tablespace named *aler_index* with at least 300M of available space.
5. Create a temporary tablespace named *aler_temp* with at least 100M of available space.

Example Tablespace Creation Script:

- ALER_DATA
- ALER_INDEX

Notice that the Tablespace scripts are for a Unix/Linux based DataFile. The file path value of *c:\oracle\...file.dbf* is appropriate syntax for Windows.

```
CREATE TABLESPACE aler_DATA
  DATAFILE '/opt/oracle/oradata/aler/aler_data.dbf' SIZE 100M AUTOEXTEND ON NEXT 10240K MAXSIZE UNLIMITED
  EXTENT MANAGEMENT LOCAL AUTOALLOCATE
  LOGGING
  ONLINE
  SEGMENT SPACE MANAGEMENT AUTO;

CREATE TABLESPACE aler_INDEX
  DATAFILE '/opt/oracle/oradata/aler/aler_index.dbf' SIZE 100M AUTOEXTEND ON NEXT 5120K MAXSIZE UNLIMITED
  EXTENT MANAGEMENT LOCAL AUTOALLOCATE
  LOGGING
  ONLINE
  SEGMENT SPACE MANAGEMENT AUTO;
```

Build a Database User

1. Log into Oracle (admin privileges required).
2. Create a user and password.
3. Ensure that the user has the following privileges:
 - Privilege to create sessions

- Ability to create tables
- Default tablespace is *aler_data*.
- Temporary tablespace is *aler_temp*.
- Assigned quota of 300M for *aler_data*.
- Assigned quota of 300M for *aler_job*.
- Assigned quota of 300M for *aler_index*.
- Assigned quota of 100M for *aler_temp*.

Example User Creation Script:

Drop user ALER cascade;

```
CREATE USER ALER IDENTIFIED BY ALER_PWD
DEFAULT TABLESPACE ALER_DATA
TEMPORARY TABLESPACE TEMP
PROFILE DEFAULT
ACCOUNT UNLOCK;
```

```
GRANT "CONNECT" TO ALER;
ALTER USER ALER DEFAULT ROLE "CONNECT", "RESOURCE";
GRANT CREATE MATERIALIZED VIEW TO ALER;
GRANT CREATE SEQUENCE TO ALER;
GRANT CREATE SESSION TO ALER;
GRANT CREATE SYNONYM TO ALER;
GRANT CREATE SNAPSHOT TO ALER;
GRANT CREATE TABLE TO ALER;
GRANT CREATE TRIGGER TO ALER;
GRANT CREATE VIEW TO ALER;
GRANT UNLIMITED TABLESPACE TO ALER;
```


SQL Server 2005 Database Installation

Appropriate administrative privileges are necessary to complete this process. These instructions assume the use of Enterprise Manager.

Prerequisites

1. A login with database administrator privileges (Usually the SA account).
2. JDBC drivers for SQL Server 2005 - *sqljdbc.jar* (driver version 1.1.1501.101)

Create a Database User

Note: This step is only necessary if you are using SQL Server Authentication for the ALER application installation.

1. Open the SQL Server 2005 Management Studio.
2. Expand the Server tree to open the **Security** folder.
3. Right-click **Logins** and select **New Login** on the pop-up menu.
4. Enter **ALER_USER** as the **Login name** (or choose a Windows login account as appropriate for your security policy).
5. Select **SQL Server Authentication**.
6. Enter a password that meets your password policy into the **Password** and **Confirm password** fields.
7. On the Select a page list, select **User Mappings** and grant database roles for *public* and *db_owner*.
8. Click **OK** to create the user account.

Create a Database

1. Open the SQL Server 2005 Management Studio.
2. Expand the server tree to open the **Databases** folder.
3. Right-click **New Database** on the pop-up menu.
4. In the New Database window, enter **ALER** as the database name.
5. Enter **ALER_USER** as the database owner (or choose a Windows login account as appropriate for your security policy).
6. Click **Add** to create a new File Group.
 - o Name the File Group **aler_index**.

- Click the **File Group** cell and use the drop-down to select **<new filegroup>**.
 - Name the new file group **INDX** and click **OK**.
7. Click **OK** again to create the database.

UDB Database Installation

Appropriate administrative privileges are necessary to complete this process. These instructions assume use of the DB2 shell program to run scripts.

Important! UDB/DB2 is not a supported database for the installation of the ALER Workflow engine.

Prerequisites

UDB database installation requires the following resources:

1. A login with database administrator privileges.
2. IBM Type 4 JDBC drivers for access to the database server.

Type 4 Drivers are packaged in *db2jcc.jar*, *db2jcc_javax.jar* and *db2jcc_license_cu.jar*. These files are normally created/distributed as a part of the DB2/UDB database server installation.

Create the Database

1. Connect to the database using a UDB authentication that allows the creation of new databases.
2. Using the following script as an example, create a database and name it **ALER**. When running the shell commands, change the example file names as appropriate.

Note: UDB database names must contain no more than eight characters). This script assumes that you are using the **db2 -tvf <sqlfile>** process.

```
--#SET TERMINATOR
-- UDB Client Invocation Command:
-- db2 -tvf (allows semicolons as the command terminator)
--
-- Dependencies:
--     UDB Client
--     In DB2 ADDNODE of server - 8 character alias (@NODE-NAME)

-- Create the ALER database
DROP DATABASE ALER;
CREATE DATABASE ALER USING CODESET UTF-8 TERRITORY US;

-- Connect to the ALER database
CONNECT TO ALER USER ALER_USER USING ALER_USER_PWD;

-- Create the ALER 32K Buffer Pools
CREATE BUFFERPOOL ALER32KDATAbp SIZE 1000 PAGESIZE 32K;
CREATE BUFFERPOOL ALER32KINDXbp SIZE 1000 PAGESIZE 32K;
CREATE BUFFERPOOL ALER32KBLOBbp SIZE 1000 PAGESIZE 32K;
CREATE BUFFERPOOL ALER32KTEMPbp SIZE 1000 PAGESIZE 32K;

CONNECT RESET;

CONNECT TO ALER USER ALER_USER USING ALER_USER_PWD;

CREATE REGULAR TABLESPACE alerdatats PAGESIZE 32 K MANAGED BY DATABASE USING(file '/opt/IBM/DB2/data/alerdatats.dat'
300M) extentsize 512k BUFFERPOOL ALER32KDATAbp;
CREATE REGULAR TABLESPACE alerindxts PAGESIZE 32 K MANAGED BY DATABASE USING(file '/opt/IBM/DB2/data/alerindxts.dat'
300M) extentsize 512k BUFFERPOOL ALER32KINDXbp;
CREATE REGULAR TABLESPACE alerlobts PAGESIZE 32 K MANAGED BY DATABASE USING(file '/opt/IBM/DB2/data/alerlobts.dat'
300M) extentsize 512k BUFFERPOOL ALER32KBLOBbp;
-- Create the ALER Temp Tablespace
CREATE TEMPORARY TABLESPACE alertempts PAGESIZE 32 K MANAGED BY DATABASE USING(file '/opt/IBM/DB2/data/alertempts.dat'
128M) BUFFERPOOL ALER32KTEMPbp;
```

```

-- TUNING PARAMETERS
-- Default Values
-- UPDATE DATABASE CONFIGURATION FOR ALER USING APP_CCTL_HEAP_SZ 128;
-- UPDATE DATABASE CONFIGURATION FOR ALER USING CHNGPGS_THRESH 60;
-- UPDATE DATABASE CONFIGURATION FOR ALER USING DBHEAP 600;
-- UPDATE DATABASE CONFIGURATION FOR ALER USING LOCKLIST 50;
-- UPDATE DATABASE CONFIGURATION FOR ALER USING LOGPRIMARY 3;
-- UPDATE DATABASE CONFIGURATION FOR ALER USING MAXAPPLS 40;
-- UPDATE DATABASE CONFIGURATION FOR ALER USING MINCOMMIT 1;
-- UPDATE DATABASE CONFIGURATION FOR ALER USING NUM_IOCLEANERS 1;
-- UPDATE DATABASE CONFIGURATION FOR ALER USING STMHEAP 2048;
-- UPDATE DATABASE CONFIGURATION FOR ALER USING DFT_DEGREE 1;

-- New Values Growth
UPDATE DATABASE CONFIGURATION FOR ALER USING LOGBUFSZ 131;
UPDATE DATABASE CONFIGURATION FOR ALER USING MAXLOCKS 60;
UPDATE DATABASE CONFIGURATION FOR ALER USING DFT_PREFETCH_SZ 32;
UPDATE DATABASE CONFIGURATION FOR ALER USING LOGFILESIZ 1024;
UPDATE DATABASE CONFIGURATION FOR ALER USING SOFTMAX 120;
UPDATE DATABASE CONFIGURATION FOR ALER USING SORTHEAP 737;
UPDATE DATABASE CONFIGURATION FOR ALER USING CATALOGCACHE_SZ 526;
UPDATE DATABASE CONFIGURATION FOR ALER USING NUM_IOSERVERS 6;
UPDATE DATABASE CONFIGURATION FOR ALER USING PCKCACHESZ 859;
UPDATE DATABASE CONFIGURATION FOR ALER USING UTIL_HEAP_SZ 39963;
UPDATE DATABASE CONFIGURATION FOR ALER USING LOGSECOND 50;

CONNECT TO ALER USER ALER USING ALER_USER_PWD;
ALTER BUFFERPOOL ALER32KDATAbp SIZE 14532;
ALTER BUFFERPOOL IBMDEFAULTBP SIZE 3633;
COMMIT;
CONNECT RESET;

```

Tomcat Application Upgrade

These instructions enable you to upgrade to ALER 3.0 on Tomcat using the BEA Products installation program.

Prerequisites

- Please reference the [Supported Configurations](#) list for the most recent information regarding your application server version.
- If the installed application server version is no longer supported by ALER, or the application server has been upgraded to a newer version, follow the application upgrade instructions for the application server side, and then continue following the database upgrade instructions within this document.
- The directory where Tomcat is installed on your system is referenced in these instructions as `CATALINA_HOME`.
- The web application context that is being upgraded is referenced in these instructions as `CATALINA_HOME/webapps/aler`.

Create an `aler_temp` Directory

On the Tomcat server, create a directory named `aler_temp`. This directory will contain the backup copies of the current ALER application. This will also serve as a temporary location for properties files, JDBC drivers, and any customized ALER items (`home.jsp` for example) that are required for the new application.

Back Up the Current ALER Installation

Navigate to the Tomcat installation location `CATALINA_HOME/webapps` and copy the existing `aler` and `aler-web` application directories into the `aler_temp` directory.

Run the BEA Products Installation Program

1. When prompted for the **Tomcat (Catalina) Home** directory, supply the directory where the previous deployment had been installed.

2. When prompted for the **Repository application name**, use the same application name that the previous deployment had used.
3. When prompted for the **Port number**, use the same port number that the previous deployment had used.

The installer will overwrite the existing deployment with the new files, thus preserving any additional custom files (such as an upload repository directory, custom home pages, etc.).

Restore the ALER Backup Files

1. Navigate to the `aler_temp` directory.
2. Copy the `aler_temp/aler/WEB-INF/classes/*.xml` files to the `CATALINA_HOME/webapps/aler/WEB-INF/classes` directory. This action overwrites the XML files that are a part of the new install.
3. Copy the `aler_temp/aler/WEB-INF/classes/*.properties` files to the `CATALINA_HOME/webapps/aler/WEB-INF/classes` directory. This action overwrites the XML files that are a part of the new install.
4. Copy the ALER 3.0 supported JDBC driver file(s) to the `CATALINA_HOME/webapps/aler/WEB-INF/lib` directory.
5. Copy the `aler_temp/aler/custom/home.jsp` (and/or all of the custom home pages) to the `CATALINA_HOME/webapps/aler/custom` directory. This overwrites the existing `home.jsp` file that is part of the new install. In addition, copy any custom images that were included in the custom home page(s) to the appropriate directory.
6. If using any custom Type icons, copy the files from the `aler_temp/aler-web/images/tree/enterprise/asset` directory to the `CATALINA_HOME/webapps/aler-web/images/tree/enterprise/asset` directory.
7. If custom reports were created, see the [Migrating Custom Reports](#) section.
8. Copy/Move the Upload Repository directory (normally found in `aler_temp/aler/custom/upload`) into the new `CATALINA_HOME/webapps/aler/custom` installation directory.
9. If the previous deployment utilized the PDF Printing application, then the XSL files that are being stored in the separate deployment will need to be moved to the `CATALINA_HOME/webapps/aler/pdfreport/xsl` directory. The PDF Printing application has been merged within ALER 3.0. Additional property changes will also need to be updated once the application is running.
10. If additional files were included or modified in the original ALER install, consult BEA Technical Support for assistance with including these files into the ALER 3.0 installation.

Start Tomcat

Important! If you are using LDAP: confirm **Enable LDAP Login** is set to **True** and verify the **LDAP Configuration**. If you are using SSO: confirm **Enable Advanced Container Login** is set to **True** and verify the **Advanced Container Configuration**. If your LDAP or SSO configurations are not set correctly, users will not be able to authenticate with the ALER application.

1. Navigate to the `CATALINA_HOME/work` directory and delete the cache for the ALER 3.0 application.
2. Start the Tomcat application server.

Test the Application

ALER 3.0 is now deployed. Testing the installation through the use of the ALER Diagnostics page is recommended. Recommended tests are:

- Product Version Information
- Test Required Libraries
- Test Database

WebLogic Server Application Upgrade

These instructions enable you to upgrade to ALER 3.0 on WebLogic Server using the BEA Products installation program.

Prerequisites

- ALER 3.0 requires WebLogic Server 9.2, Maintenance Pack 1. If the currently installed ALER application is not running on the required version, upgrade the application server and the existing ALER domain appropriately.
- If the application server version in use is no longer supported, or if the application server has been upgraded to a new version, follow the new application instructions for the application server side, and then continue following the database upgrade instructions in this section.
- It is assumed that the ALER domain is installed within the default location of `BEA_HOME/user_projects/applications/aler`. If the ALER domain is installed in a different location, then make the appropriate adjustments to these instructions.

Create ALER Temp Directories

- On WebLogic Server, create a directory named `aler_upgrade`. This directory is used to contain the ALER upgrade files.
- On WebLogic Server, create a directory named `aler_temp`. This directory is used to backup the current ALER application. This location is used as a temporary location for properties files, JDBC drivers, and any customized ALER items (`home.jsp`, for example) that are necessary for the new application.

Back Up the Current ALER Installation

1. Log into the WebLogic Server Administration Console and select the ALER deployment.
2. Choose the option to `Lock & Edit`
3. Choose the option to stop the application and confirm the choice if required.
4. Select the ALER application and choose the option to uninstall the ALER deployment. This

operation will remove the deployment from the domain's config.xml file without removing any of the files within the deployment. The ALER Product Installer application will create a new domain for the ALER 3.0 installation. Therefore the existing domain will no longer be used.

5. Navigate to the `BEA_HOME/user_projects/applications/aler` directory and copy the `aler-app` and `aler-web` directories to the *aler_temp* directory.

Remove Current ALER Installation Files

Caution! Perform this step only after completing backup of the directory, as described in [Back Up the Current ALER Installation](#).

Remove the ALER application `BEA_HOME/user_projects/applications/aler/aler-app`, `BEA_HOME/user_projects/applications/aler/aler-app` and `BEA_HOME/user_projects/applications/aler/META-INF` directories.

Run the BEA Products Installation Program

1. When prompted for the **BEA Home** directory, supply the directory where the WebLogic Server 9.2 MP1 is installed.
2. When prompted for the **Domain name** select the same domain name that had been deployed previously.
3. When prompted for the **Fully qualified server name**, supply the fully qualified host name of the server that this domain is being installed on.
4. When prompted for the **Repository application name**, use the same application name that the previous deployment had used.
5. When prompted for the **Port number**, use the same port number that the previous deployment had used.

ALER and ALBPM Workflow Database Installation

1. If you plan to install the Workflow option as a part of this upgrade, then when prompted to configure the database, choose **Yes**. This will be a required process for the Workflow deployment. Otherwise, you may choose the **No** option to bypass this and the following step.
2. A second ALER database will be necessary to allow the BEA Products installer to pass to the Workflow database and application configuration stage. Once this process is completed, this second database can be dropped without impact on the upgraded deployment.

Restore the ALER Backup Files

1. Navigate to the `aler_temp` directory.
2. Copy the `aler_temp/aler-app/WEB-INF/classes/*.xml` files to the `BEA_HOME/user_projects/domains/<domain>/aler-app/WEB-INF/classes` directory. This action overwrites the XML files that are a part of the new install.
3. Copy the `aler_temp/aler-app/WEB-INF/classes/*.properties` files to the `BEA_HOME/user_projects/domains/<domain>/aler-app/WEB-INF/classes` directory. This action overwrites the XML files that are a part of the new install.
4. Copy the ALER 3.0 supported JDBC driver file(s) to the `BEA_HOME/user_projects/domains/<domain>/aler-app/WEB-INF/lib` directory.
5. Copy the `aler_temp/aler/custom/home.jsp` (and/or all of the custom home pages) to the `BEA_HOME/user_projects/domains/<domain>/aler-app/custom` directory. This overwrites the existing `home.jsp` file that is part of the new install. In addition, copy any custom images that were included in the custom home page(s) to the appropriate directory.
6. If using any custom Type icons, copy the files from the `aler_temp/aler-web/images/tree/enterprise/asset` directory to the `BEA_HOME/user_projects/domains/<domain>/aler-web/images/tree/enterprise/asset` directory.
7. If custom reports were created, see the [Migrating Custom Reports](#) section.
8. Copy/Move the Upload Repository directory (normally found in `aler_temp/aler/custom/upload`) into the new `BEA_HOME/user_projects/domains/<domain>/aler-app/custom` installation directory.
9. If the previous deployment utilized the PDF Printing application, then the XSL files that are being stored in the separate deployment will need to be moved to the `BEA_HOME/user_projects/domains/<domain>/aler-app/pdfreport/xsl` directory. The PDF Printing application has been merged within ALER 3.0. Additional property changes will also need to be updated once the application is running.
10. If additional files were included or modified in the original ALER install, consult BEA Technical Support for assistance with including these additions into ALER 3.0 installation.

Start WebLogic Server

Important! If you are using LDAP: confirm **Enable LDAP Login** is set to **True** and verify the **LDAP Configuration**. If you are using SSO: confirm **Enable Advanced Container Login** is set to **True** and verify the **Advanced Container Configuration**. If your LDAP or SSO configurations are not set correctly, users will not be able to authenticate with the ALER application.

1. Delete the application server cache located within `ALER_DOMAIN/servers/AdminServer/tmp/WL_user/aler` directory. This directory contains compiled JSP pages that pertain to the old application deployment, which are no longer valid.
2. Start the application through the WebLogic Server Administration Console.

Test the Application

ALER 3.0 is now deployed. Testing the installation through the use of the ALER Diagnostics page is recommended. Recommended tests are:

- Product Version Information
- Test Required Libraries
- Test Database

WebSphere Application Upgrade

These instructions enable you to upgrade to ALER 3.0 on WebSphere using the BEA Products installation program.

Prerequisites

- ALER 3.0 requires WebSphere version 6.1.0.5. If the currently installed ALER application is not running on the required version, upgrade the application server as appropriate.
- If the application server version on which the currently installed version of Flashline/ALER is running is no longer supported, or if the application server has been upgraded to a new version, follow the application instructions for the application server side, and then continue following the database upgrade instructions in this section.
- The current deployment directory containing the ALER application is referred to as `aler_current` (this directory will normally be located in the following directory: `WAS_HOME/AppServer/profiles/AppSrv01/installedApps/localhostNode01Cell/installedApps/aler.ear`).
- Once the upgrade process is completed, the ALER property, XML and JDBC driver files will be located within a new `BEA_HOME/repository30/websphere` directory.

Create an `aler_temp` Directory

On the WebSphere server, create a directory named `aler_temp`. This directory will contain backups of the current ALER application properties files, JDBC drivers, and any customized AquaLogic Enterprise Repository items (`home.jsp` for example) that are necessary for the new application.

Back Up the Current ALER Installation

1. Copy the contents of the current `ALER_HOME` shared library directory contents to the `aler_temp` directory. The `aler_temp` directory should then contain all of the original ALER properties files, XML files, and your current JDBC driver JAR files.
2. If you have a customized ALER home page, navigate to the `aler_current/aler-app.war/custom` directory and copy the `home.jsp` file to the `aler_temp` directory. If you have used the Role or Department based home pages, copy those files from the directory that is indicated within the `aler_current/aler-app.war/WEB-INF/classes/homepages.xml` file.

3. Copy any custom images that are used in the custom home page file(s) to the `aler_temp` directory.
4. If you have any custom Asset Type icons, navigate to the `aler_current/aler-web.war/images/tree/enterprise/asset` directory and copy the images to the `aler_temp` directory.
5. Optionally, copy all XSLT templates from the PDF Printing web-application to the `aler_temp` directory. This web application is bundled directly into the core of the ALER product.
6. Copy any additional files or directories that were included or modified from the base ALER installation into the `aler_temp` directory.

Remove the Current ALER Installation Files

1. Start the WebSphere 6.1.x Administrative Console.
2. Expand the Applications -> Enterprise Applications folder and select the **ALER** check box.
3. Click **Stop**.
4. Select the **ALER** check Box.
5. Click **Uninstall**.
6. Repeat the above steps to remove the PDF Printing web-application (if present). This feature is now bundled within the ALER base application.
7. Save the changes to the Master Configuration when prompted.

Run the BEA Products Installation Program

1. When prompted for the **Repository application name**, use the same application name that the previous deployment had used.
2. When prompted for the **Port number**, use the same port number that the previous deployment had used.

The BEA Products installer will generate a deployable EAR file, thus allowing you to quickly and easily install the updated version of the application directly from the application server's file system. It will also create and deploy the ALER Workflow application components appropriately.

Install New Application Files

1. Start the WebSphere Administrative Console.
2. Expand the Applications -> Enterprise Applications folder and click **Install**.
3. Choose the option to select a **Remote Filesystem** (this option is relative to the application server), browse to the `BEA_HOME/repository30/websphere` directory, and select the `aler.ear` file.
4. Select the **Prompt me only when additional information is required** option and click **Next** to continue.
5. On the Select Installation Options page, click **Next** to continue after making the appropriate changes.
6. On the Map modules to servers page, click **Next** to continue after making the appropriate changes.
7. On the Map virtual hosts for Web modules page, click **Next** after making the appropriate changes.
8. On the Deployment summary page, click **Finish** to deploy the EAR file.

Note: This process may take several minutes.

9. After the installation is finished, the **Application installed successfully** message is displayed.
10. Click **Save to Master Configuration** when prompted.

Restore the Backup ALER Files

1. If you have a customized ALER home page(s), copy `home.jsp` to the `aler_current/aler-app.war/custom` directory. This operation will overwrite the `home.jsp` file that is part of the new install.
2. Copy any custom images that may be part of the `home.jsp` file to the appropriate directory.
3. If using any custom Type icons, copy the files to the `aler_current/aler-web.war/images/tree/enterprise/asset` directory.
4. If custom reports were created, see the [Migrating Custom Reports](#) section.
5. If additional files were included or modified in the original ALER install, consult BEA Technical Support for assistance with inclusion of these additions into ALER 3.0.
6. Copy all previous property files from the `aler_temp` to the `BEA_HOME/repository30/websphere` directory.
7. Copy all previous XML files from the `aler_temp` to the `BEA_HOME/repository30/websphere` directory.
8. Copy the currently supported JDBC Driver files into the `BEA_HOME/repository30/websphere`

directory.

9. Copy existing XSTL files for PDF Printing output into the *aler_current/aler-app.war/pdfreport/xsl* directory.

Configure the WebSphere Settings

1. Start the Administrative Console.
2. Expand the Applications -> Enterprise Applications folder and select the **ALER** application.
3. Click the **Manage Modules** link.
4. Select the **aler-app.war** module.
5. In Class Loader order, change **Classes loaded with parent class loader first** to **Classes loaded with the application class loader first**.
6. Click **OK**.
7. Select the **aler-web.war** module.
8. In Class Loader Order, change **Classes loaded with parent class loader first** to **Classes loaded with the application class loader first**.
9. Click **OK**.
10. Click **Save**.

Add a Shared Library to the Application Server

1. Expand the **Environment** folder and click the **Shared Libraries** link.
2. Select the scope of the Shared Library and click **New**.
3. Give the shared library a meaningful name, such as *ALER_3.0*.
4. Add the CLASSPATH for the JDBC driver and application properties file(s) located in the *aler_home* directory (one entry per line is all that is required). The following are CLASSPATH examples:
 - o Windows OS:
 - *c:\bea\repository30\websphere* (this directory contains the .properties, XML and .DAT files for the ALER application)
 - *c:\bea\repository30\websphere\ojdbc14.jar* (for oracle 10g support)
 - *c:\bea\repository30\websphere\sqljdbc.jar* (for SQL Server 2005 support)
 - *c:\bea\repository30\websphere\db2jcc.jar* (for DB2/UDB support)

- c:\bea\repository30\websphere\db2jcc_javax.jar (for DB2/UDB support)
 - c:\bea\repository30\websphere\db2jcc_license_cu.jar (for DB2/UDB support)
- Linux/Unix OS:
- /opt/bea/repository30/websphere (this directory contains the .properties, XML and .DAT files for the ALER application)
 - /opt/bea/repository30/websphere/ojdbc14.jar (for oracle 10g support)
 - /opt/bea/repository30/websphere/sqljdbc.jar (for SQL Server 2005 support)
 - /opt/bea/repository30/websphere/db2jcc.jar (for DB2/UDB support)
 - /opt/bea/repository30/websphere/db2jcc_javax.jar (for DB2/UDB support)
 - /opt/bea/repository30/websphere/db2jcc_license_cu.jar (for DB2/UDB support)
5. Click **Apply**.
 6. Click **Save** directly to Master Configuration link.
 7. Associate the ALER Web Modules with the new Shared Library.
 8. Select the ALER application the Applications -> Enterprise Applications -> ALER -> **Shared Library references** link.
 9. Select the **aler-app.war** module from the list of web modules and click **Reference Shared Libraries**.
 10. Select the **ALER_3.0** shared library in the Available list box, and use the >> button to select that library for use with the application.
 11. Click **OK** to continue.
 12. Select the **aler-web.war** module from the list of web modules and click **Reference Shared Libraries**.
 13. Select the **ALER_3.0** shared library in the Available list box, and use the >> button to select that library for use with the application.
 14. Click **OK** to continue.
 15. Follow this process to add any additional libraries that may be present for JDBC drivers or other required libraries for the application.
 16. Click the **Save to master config** when prompted.

Note: The CLASSPATH of the server can be set within the cell's **server.xml** file if the use of a shared library is not possible: (<server_path>/AppServer/config/cells/<cell_name>/nodes/<node_name>/servers/server1.xml). This file contains an element named jvmEntries that may also include a CLASSPATH element. If no CLASSPATH element is currently defined, one can added manually. For example: <classpath>/opt/bea/repository30/websphere:/opt/bea/repository30/websphere/ojdbc14.jar</classpath>. A change of this nature will require that a

server restart.

Start WebSphere

Important! If you are using LDAP: confirm **Enable LDAP Login** is set to **True** and verify the **LDAP Configuration**. If you are using SSO: confirm **Enable Advanced Container Login** is set to **True** and verify the **Advanced Container Configuration**. If your LDAP or SSO configurations are not set correctly, users will not be able to authenticate with the ALER application.

1. Navigate to the `WebSphere_HOME/AppServer/temp/` directory and delete the cache for the ALER application.
2. Restart WebSphere.

Test the Application

ALER 3.0 is now deployed. Testing the installation through the use of the ALER Diagnostics page is recommended. Recommended tests are:

- Product Version Information
- Test Required Libraries
- Test Database

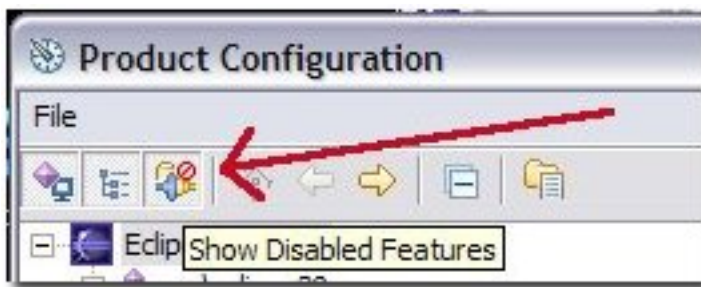
Migrating Custom Reports

Editing the CustomReports.xml file

1. Navigate to the `./WEB-INF/config/reports/` directory on the app server.
2. Open the `customreports.xml` file in a text editor.
3. Edit the meta-tags for **report name**, **display name**, **description**, and **external** to reference a custom report.

ALER Legacy Eclipse Plug-in Uninstallation

1. In Eclipse, click **Help**
2. Click **Software Updates**
3. Click **Manage Configuration**
4. Select **Show disabled features**.



5. Select the **AquaLogic Enterprise Repository** feature.
 6. Click **disable feature**.
- Note: Unless the feature is properly disabled prior to uninstallation, some AquaLogic Enterprise Repository files may remain in the Eclipse plugins directory. These files must be manually deleted.
7. Follow the prompts to restart Eclipse.
 8. After restart, select **Manage Configuration** in the **Help** menu.
 9. Select the **AquaLogic Enterprise Repository** feature.
 10. Select **Uninstall**.
 11. Follow the prompts to restart Eclipse.

Verifying the Assets In Progress Property Value

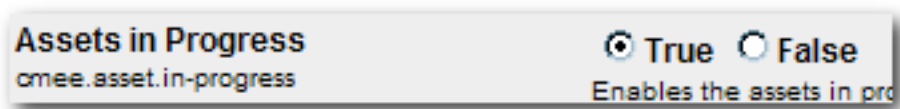
Two properties must be enabled in order to activate **Assets in Progress**. This procedure is performed on the ALER **Admin** screen. This property **MUST** be enabled when using the WorkSpace Studio with the ALER product. (For more information on using WorkSpace Studio with ALER, refer to the [Prerequisites for Using WorkSpace Studio With ALER](#) instructions.

If you are upgrading an existing instance of Flashline/ALER and the Assets In Progress feature had not been enabled, then this feature should be disabled **PRIOR** to any users accessing the instance and modifying assets records.

Enabling Assets in Progress

1. Click **System Settings**.
2. Click **General Settings** in the **System Settings** section.
3. Enter the property `cmee.asset.in-progress` in the **Enable New Property** text box.
4. Click **Enable**.

Assets in Progress appears in the list of properties in the main pane.



5. Make sure the property is set to **True**.
6. Click **Save**.
7. Enter the property `cmee.asset.in-progress.visible` in the **Enable New Property** text box.
8. Click **Enable**.

View Assets in Progress appears in the list of properties in the main pane.



9. Make sure the property is set to **True**.
10. Click **Save**.

The **Registration Status** drop-down menu will now appear in the **Search** section on the AquaLogic Enterprise Repository **Assets** screen.

Disabling Assets in Progress

This procedure is performed on the AquaLogic Enterprise Repository **Admin** screen.

Note: Disabling the **Assets in Progress** feature prevents **Asset Editor** access to assets in the **Unsubmitted** state. Assets in the **Submitted - Pending Review** state will remain accessible to the **Asset Editor**. Before disabling **Assets in Progress** it is recommended that all necessary assets be promoted from **Unsubmitted** to **Submitted - Pending Review**.

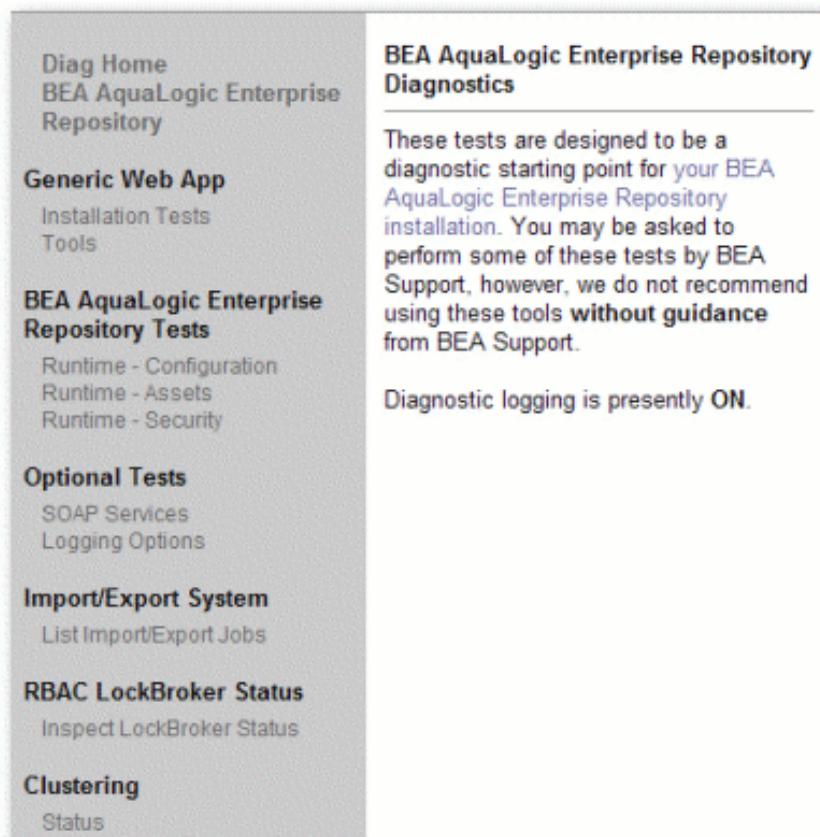
1. Click **System Settings**.
2. Click **General Settings** in the **System Settings** section.
3. Locate **Assets in Progress** in the list of properties in the main pane.
4. Set the **Assets in Progress** property to **False**.
5. Locate **View Assets in Progress** in the list of properties in the main pane.
6. Set the **View Assets in Progress** property to **False**.
7. Click **Save**.

Utilizing the Diagnostics Testing Tool

The **Diagnostics** tool allows testing and troubleshooting of certain aspects of BEA AquaLogic™ Enterprise Repository application.

Launching the Diagnostic Tool

To launch the ALER **Diagnostics** tool, navigate to: `http://host_name/application_name/diag/index.jsp`



Navigating the Diagnostics Tool

In addition to the **Diag Home** and **BEA AquaLogic Enterprise Repository** links, the left sidebar of the **Diagnostics** tool (the gray area in the image above) includes links to several tests that can be run to check your installation of ALER. Click any of these links to display the available tests in the main pane of the **Diagnostics** tool.

- **Generic Web App**
 - Installation Tests
 - *Product Version Information*
 - Provides information on the installed version of the ALER, the version of the installed ALER database, upgrade dates, and the maintenance mode setting for the database.
 - *Snoop Servlet*

- Provides an output listing of all cookies, headers, CGI-Variables, Servlet Context Attributes, Session information and Session values set as a part of the request to the application server.
- *Test Required Libraries*
 - Provides information about the application server's configuration regarding required JAR library files, installed XML parsers, and the presence of the minimally required ALER properties files.
- *Run Active Diagnostic Tests*
 - Runs a JUnit test against the installed ALER application. Many of these tests are required to allow ALER to build and parse XML data.
- *Test Database*
 - Provides insight into the JDBC driver functionality, including the number of database side functions that are expected to be supported. Also tests the connection string set in the `database.properties` file, provides additional information regarding the version of the JDBC driver, and may also indicate deficiencies within the database tables.
- *List System Paths*
 - Lists the pathways (local and absolute URL's) configured within the `cmee.properties` file as well as those stored within the database. Also helps to indicate why some parts of the application deployment function while others do not (such as images not loading).

Installation Tests	
Product Version Information	Displays internal product version, database schema version, and information about the most recent upgrade/install performed to BEA AquaLogic Enterprise Repository.
Snoop servlet	Tests servlet engine response, shows parameters passed from web client to server.
Test Required Libraries	Tests libraries, property files and system properties essential to proper functionality.
Run Active Diagnostic Tests	Performs tests against common application and environment troublespots.
Test Database	Tests database container access, availability of JDBC driver and presence of important BEA AquaLogic Enterprise Repository data tables.
List System Paths	Lists all BEA AquaLogic Enterprise Repository paths. Similar to 'List Enabled System Settings' below, but limited to world-visible paths and therefore does not require login.
Test Browser Redirection Compatibility	Via Response Code 302 - to standard URL Via Response Code 302 - to local OS file

○ Tools

- *JSP Precompiler*
 - Forces the application server to perform an immediate JIT compile of JSP pages within the ALER application. This process greatly improves the performance of page loads if performed after each application server restart.
- *Encrypt Strings for Passwords*
 - Performs a Hash on a clear-text password string supplied in the text field. This feature is used for encrypting passwords to be stored within application properties files. This tool should not be used to encrypt user passwords. Please contact BEA Support for the proper procedure to update a password in the database.

Tools

JSP Precompiler	Pre-compiles most JSPs in the system for speedier first time access. This is likely to take quite awhile (20-30 minutes), and generates false errors as JSPs are loaded without the proper parameters passed. This is OK and expected.
Encrypt Strings for passwords	Converts strings to encrypted passwords for use in misc. places in the system (see documentation for more details)

- **BEA AquaLogic Enterprise Repository Tests**

- Runtime - Configuration
 - *List Enabled System Settings*
 - Lists all values set in the ALER database as well as those set in properties files.
 - *Edit System Paths*
 - Allows an administrator to make permanent changes to the `cmee.properties` file on the application server.

Runtime - Configuration

These tests require either a BEA AquaLogic Enterprise Repository login cookie to be set on your browser, or for you to have been logged into BEA AquaLogic Enterprise Repository recently with your browser enabled to allow session cookies.

List Enabled System Settings	Lists all enabled BEA AquaLogic Enterprise Repository settings. Values that deviate from the default are colored blue.
Edit System Paths	Edits BEA AquaLogic Enterprise Repository paths. This can be used to bootstrap a new installation.
Custom Home Pages	Current Home Page selection Criteria

- Runtime - Assets
 - *List Types*
 - Produces XML output that describes all Asset Types stored within the ALER database.
 - *Edit Asset Custom Data*
 - Allows a registrar to modify XML data contained within an asset in the event that an asset's metadata is otherwise uneditable. **Do not use this tool unless specifically directed by BEA Support personnel**
 - *Get XML: Show Type ID*
 - Produces the XML description of a specific Asset Type based on the unique ID of the Asset Type record.
 - *Get XML: Type ID As String*
 - Produces the XML description of a specific Asset Type based on the unique ID of the Asset Type record as one continuous string.
 - *Get XML: Type XML Schema*
 - This tool produces the XML schema of a specific Asset Type based on the unique ID of the Asset Type record.
 - *Get XML: Type XML Schema as String*
 - This tool produces the XML schema of a specific Asset Type based on the unique ID of the Asset Type record as one continuous string.
 - *Get XML: Asset ID*
 - Produces the XML data of an Asset based on the unique ID of the Asset.
 - The **include extended metadata** option produces additional information related to Policies and other data.
 - *Get XML: Asset ID as String*
 - Produces the XML data of an Asset based on the unique ID of the Asset as one continuous string.

Runtime - Assets

These tests require either a BEA AquaLogic Enterprise Repository login cookie to be set on your browser, or for you to have been logged into BEA AquaLogic Enterprise Repository recently with your browser enabled to allow session cookies.

List Types	Lists all types in the system. The ID's provided here are useful in the next test.
Edit Asset Custom Data	Allows editing the custom data XML for an asset. This is useful when fixing invalid xml that is preventing an asset from loading.
Search	Submits a search using criteria formatted in XML.

Get XML

Show Type ID: <input type="text"/> Editor Show	Displays the XML for the specified type id. You may need to right click and select View Source to see the XML.
Show Type ID as String: <input type="text"/> Editor Show	Displays the XML as a single string for the specified type id.
Show Type XML Schema: <input type="text"/> Show	Displays the XML Schema for an type. You may need to right click and select View Source to see the XML.
Show Type XML Schema as String: <input type="text"/> Show	Displays the XML Schema for an type as a single string for the specified type.
Show Asset ID: <input type="text"/> <input type="checkbox"/> Include extended metadata. Show	Displays the XML for the specified asset id. You may need to right click and select View Source to see the XML.
Show Asset ID as String: <input type="text"/> Show	Displays the XML as a single string for the specified asset id.
Show Asset ID extended metadata entries: <input type="text"/> Show	Itemizes the extended metadata entries for the specified asset id.

Runtime - Security

Runtime - Security

These tests require either a BEA AquaLogic Enterprise Repository login cookie to be set on your browser, or for you to have been logged into BEA AquaLogic Enterprise Repository recently with your browser enabled to allow session cookies.

- [List Auth Tokens](#)
- [List Custom Access Settings](#)

Optional Tests

- [SOAP Services](#)

SOAP Services

These tests are not always necessary for BEA AquaLogic Enterprise Repository to function properly. Run these if instructed to do so by BEA Support.

Soap Admin Console	Loads the soap admin console. List should show forum and openapi services if configured.
--------------------	--

- Logging Options

Logging Options

These tests are not always necessary for BEA AquaLogic Enterprise Repository to function properly. Run these if instructed to do so by BEA Support.

Edit Logs	Allows changing of different logging options while registry is running.
-----------	---

- Import/Export System
 - List Import/Export Jobs

Import/Export Jobs Currently Active in System

ID	Type	Description	State	Complete	Start Date	Last Update	Owner	Operations
----	------	-------------	-------	----------	------------	-------------	-------	------------

- RBAC LockBroker Status
 - Inspect LockBroker Status

Currently Held Locks

Recently Failed Lock Requests

- To access the ALER login page click the **AquaLogic Enterprise Repository** link in the sidebar of the **Diagnostics** page.
- To activate the debugging mode for the current session, click the **turning logging on** link in the main pane.

Checking Product Version Information

- Click **Installation Tests** under **Generic Web App**.
- Click **Product Version Information** in the main pane. .

A new browser window opens to display information on the internal product version, database schema version, and information about the most recent upgrade/install.

BEA AquaLogic Enterprise Repository Version Information

Product Version: 2.5.0

Database Version: 14.0.0

BEA AquaLogic Enterprise Repository database version properly matches the running instance of ALER.

Information From Database:

Database Installed From: 5.2.0

Database Version: 14.0.0

Database Date: 2006-06-02 19:06:20.0

Upgrade Maintenance Mode: off

Steps of last upgrade:

Step	Start Date	End Date	Required?
Disable/drop Foreign Keys	2006-06-02 19:06:20.0	2006-06-02 19:06:20.0	yes
Pre-Data Schema Changes	2006-06-02 19:06:20.0	2006-06-02 19:06:20.0	yes
Data Migration	2006-06-02 19:06:20.0	2006-06-02 19:06:20.0	yes
Post-Data Schema Changes	2006-06-02 19:06:20.0	2006-06-02 19:06:20.0	yes
Java Based Migration	2006-06-02 19:06:22.0	2006-06-02 19:06:23.0	yes
Deprecation Scripts	2006-06-02 19:06:23.0	2006-06-02 19:06:23.0	no

Test Servlet Functionality

1. Click **Installation Tests** under **Generic Web App**.
2. Click **Snoop Servlet** in the main pane.

A new browser window opens to display information on the parameters passed from the Web client to server.

Snoop Servlet - Request/Client Information

Requested URL:

http://qa-app2.flashline.com:3080/flashline/Snoop

Request Information:

Request method	GET
Request URI	/flashline/Snoop
Request protocol	HTTP/1.1
Servlet path	/Snoop
Path info	<none>

Test Required Libraries

1. Click **Installation Tests** under **Generic Web App**.
2. Click **Test Required Libraries** in the main pane.

A new browser window opens to display information on the property files and system properties essential to proper functionality.

BEA AquaLogic Enterprise Repository Library Diagnostic

Examining webapp configuration

Needed Components

Found BEA AquaLogic Enterprise Repository

(com.flashline.cmee.jsp.CMEESession)
at /usr/local/build-jakarta-tomcat-5.0.25/webapps/flashline/WEB-INF/lib/cmee.jar

Found BEA AquaLogic Enterprise Repository DBAccess

(com.flashline.db.DBConnection)
at /usr/local/build-jakarta-tomcat-5.0.25/webapps/flashline/WEB-INF/lib/dbaccess.jar

Found Apache Xerces XML Parser

(org.apache.xerces.parsers.DOMParser)

Test Database Connectivity

1. Click **Installation Tests** under **Generic Web App**.
2. Click **Test Database** in the main pane.

A new browser window opens to display information on container access, the availability of JDBC drivers and the presence of important ALER data tables.

```
Self test: com.flashline.cmee.servlet.diag.Database
JDBC Driver      oracle.jdbc.driver.OracleDriver
URL              jdbc:oracle:thin:@qa-app2.flashline.com:1521:sid1
Username         cmee50qa1
Password         (private)
Driver Name      Oracle JDBC driver
Driver Version   9.2.0.5.0
String Functions ASCII,CHAR,CONCAT,LCASE,LENGTH,LTRIM,REPLA
Date Functions   CURDATE,CURTIME,DAYOFMONTH,HOUR,MINUTE
Numeric Functions ABS,ACOS,ASIN,ATAN,ATAN2,CEILING,COS,EXP,FI
System Functions USER
SQL: select count(*) as bad_editor_xmls from assetypeeditorxmls where editorx
BAD_EDITOR_XMLS
0
```

List System Paths

1. Click **Installation Tests** under **Generic Web App**.
2. Click **List System Paths** in the main pane.

A new browser window opens to display all ALER paths.

```
cmee.server.paths.image=http://qa-app2.flashline.com:3080/flashline-web/images
cmee.server.paths.jsp-tool=http://qa-app2.flashline.com:3080/flashline-
web/webstart
cmee.server.paths.jsp=http://qa-app2.flashline.com:3080/flashline
cmee.server.paths.resource=http://qa-app2.flashline.com:3080/flashline-web
cmee.server.paths.servlet=http://qa-app2.flashline.com:3080/flashline
cmee.server.paths.upload=/usr/local/build-jakarta-tomcat-
5.0.25/webapps/flashline/WEB-INF/upload
cmee.server.paths.upload-registrar=\\qa-app2\upload-autobuild\
cmee.server.paths.upload-repository=
```

Run Pre-Compile Servlet

1. Click **Tools** under **Generic Web App**.
2. Click **JSP Precompiler** in the main pane.

A new browser window opens to display a list of system JSPs that have been pre-compiled for speedier first time access.

Base URL: http://qa-app2.flashline.com:3080/flashline/

Compiling jsps using jsplist.txt...

```
Starting : http://qa-app2.flashline.com:3080/flashline/c
Finishing: http://qa-app2.flashline.com:3080/flashline/c
Starting : http://qa-app2.flashline.com:3080/flashline/i
Finishing: http://qa-app2.flashline.com:3080/flashline/i
Starting : http://qa-app2.flashline.com:3080/flashline/c
SKIPPED: http://qa-app2.flashline.com:3080/flashline/cme
Finishing: http://qa-app2.flashline.com:3080/flashline/c
Starting : http://qa-app2.flashline.com:3080/flashline/c
```

- **Note:**
 - This servlet produces errors within the error log; please reset logs before starting the application server.

Other Features

Additional **Diagnostics** functionality includes:

- Testing browser redirection compatibility to both a standard URL or local file
- Listing all enabled ALER Settings
- Listing all Types in the system

- Displaying XML for a specific Type (both editor and viewer).
- Displaying ml for a specific asset.
- Current `authtokens` and custom access settings.
- Listing import/exports jobs
- Inspecting LockBroker status
- SOAP admin console

You may be instructed by BEA Support to use these optional features as a troubleshooting measure.

Using an X Windows Server with the ALER Reporting Engine

When running on Unix/Linux, the reporting engine in AquaLogic Enterprise Repository requires a connection to an X Windows server in order to generate reports. This is a functional requirement of the Java JRE, which creates a handle to the graphics environment to determine font sizes and perform other graphics-related operations. On Windows platforms, availability of the graphics environment is a given, but on Linux and other Unix platforms, it is not always present. The X Windows server can run on either the same machine as the application server or on another machine, provided that proper access is granted to use the X Windows environment from the application server.

Two steps are necessary for use (unless the application server is started while logged into the X Windows environment). The X Windows security model generally only allows applications to use the environment if they are started within the X Windows environment.

Granting Temporary Access to an X Windows server

1. On a local machine, to allow a background server process access to the X Windows server, perform the following command in an xterm window (or equivalent):
 - o `xhost +<user>`
 - (Where <user> is the username of the application server process.)
2. To allow the application to connect to a remote server, perform the following command on the remote X Windows server in an xterm window (or equivalent):
 - o `xhost +<ip>`
 - (Where <ip> is the IP address of the application server.)

Granting Permanent Access to an X Windows server

To configure X Server access to selected hosts follow these three steps:

1. Search through all files in the directory `/usr/lib/X11/xdm` (or in `/etc/X11/xdm`) for occurrences of the command `"xhost +"` or `"/usr/bin/X11/xhost +"`.
2. Remove or comment out all such lines.
3. Create or edit the file `"/etc/Xn.hosts"` where 'n' is the display number of the server on the local host, normally 0, as in `"/etc/X0.hosts"`.
4. To deny all X access to your system, the file `/etc/X0.hosts` will contain a single character, `"-"`.
5. To grant access to hosts `"localhost"`, `"newhost.gov"` and `"secondhost.gov"` and no other hosts the file `/etc/X0.hosts` will consist of:
 -
 - +localhost
 - +newhost.gov
 - +secondhost.gov

Setting DISPLAY Variable

1. On the application server, preferably in the application server software's startup script, add the following (assumes an *sh* or *sh* derivative shell):
 - `DISPLAY=<xwindows host>:0.0 export DISPLAY,`
 - (Where `<xwindows host>` is the name or IP of the remote X Windows server.)
 - (`localhost` may be used for a server running on the same machine)
2. Restart the application server.

Other Notes

- While the reporting engine does not require the X Windows server to run on the application server itself, the X11 client libraries must be accessible on the application server to reference the graphics toolkit functions.
- In order to test the authorization and setting of the `DISPLAY` variable, launch an application, such as **xterm** or **xeyes** from the application server and verify that the application appears properly on the remote X Windows desktop.
- If you experience authorization problems with the application server connecting to the remote X Windows server, try "`xhost +`" to allow all remote client access to the X Windows server.

X Windows Alternatives

For installations where X Windows is not a feasible option within the Enterprise Architecture, or where servers are racked together within a datacenter and run headless. Since each of these options simulates an X server, the X Windows installation instructions above regarding setting the appropriate `DISPLAY` environment variable and the use of the `xhost` commands also apply. Two options have proven to be effective and compliant within many environments:

- **XVfb** - X Virtual Frame Buffer - An alternative to a fully fledged X Windows installation.
- **VNC Server** - Again an effective alternative with added benefits of allowing a remote console connection to the hosting machine if configured properly.
 - One of the more common misconfigurations with the VNC Server option is how a firewall will block a listening port for either of these services. For example, `vncserver` by default will listen on TCP port 5901 which would correspond with a `DISPLAY=:1.0`. A `vncserver` process configured to run on `DISPLAY=:11.0` the process would be listening on TCP port 5911, etc.
 - The very minimal installation of `vncserver` is all that is required to allow the functionality of the AquaLogic Enterprise Repository reporting engine.