Oracle® Fusion Middleware

Installation and Upgrade Guide for Oracle Identity Analytics

11*g* Release 1, Patch Set 1 (11.1.1.5)

E23378-03

March 2014

ORACLE®

Oracle Fusion Middleware Installation and Upgrade Guide for Oracle Identity Analytics 11g Release 1, Patch Set 1 (11.1.1.5)

E23378-03

Primary Author: Deena Purushothaman

Contributing Authors: Kevin Kessler, Neha Sethi

Contents

9 Upgrading Oracle Identity Analytics in a Production Environment

Preface

This guide describes how to install Oracle® Identity Analytics software as well as how to upgrade from an older version of Sun™ Role Manager.

Audience

This guide is intended for system administrators, system deployers, database administrators, and system integrators who need to install or upgrade the Oracle Identity Analytics software.

To be successful with this guide you should have a solid understanding of your application server software and your database server software, as well as an advanced knowledge of J2EE principles.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Documents

For more information, see the following documents in the Oracle Identity Analytics Release 11g R1 PS1 documentation set:

- *Oracle Identity Analytics Release Notes*
- *Oracle Identity Analytics Administrator's Guide*
- *Oracle Identity Analytics User's Guide*
- *Oracle Identity Analytics System Integrator's Guide*
- *Oracle Identity Analytics API Guide*
- *Oracle Identity Analytics Database Administrator's Guide*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Part I

Compatibility Matrix

Part I is the compatibility matrix, which shows the products that the Oracle® Identity Analytics 11*g*R1 software supports.

Part I contains the following chapters:

1

Oracle Identity Analytics 11g R1 PS1 Compatibility Matrix

This chapter contains the following sections:

For information about supported application servers, databases, web servers, and so on, see the platform certification matrix:

http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-certification-100350.html

1.1 Supported Languages

The Oracle Identity Analytics 11gR1 PS1 application supports the following languages:

- Simplified Chinese
- Traditional Chinese
- English
- French
- German
- Italian
- Japanese
- Korean
- Brazilian Portuguese
- Spanish

Part II

Installing Oracle Identity Analytics

Part II contains instructions on how to install the Oracle Identity Analytics software.

Part II contains the following chapters:

2

Preparing to Install Oracle Identity Analytics

This chapter contains the following sections:

Complete the steps in this chapter prior to beginning the steps in Chapter 3, "Configuring Your Oracle Identity Analytics Installation Prior to Deployment."

2.1 Architecture Overview

Oracle Identity Analytics is a Java™ 2 Platform, Enterprise Edition (J2EE platform) web application. The J2EE platform consists of a set of industry-standard services, APIs, and protocols that provide the functionality for developing multitiered, web-based, enterprise applications. The Oracle Identity Analytics system architecture is distributed across three logical tiers:

- The presentation tier
- The logic tier
- The data tier

The Oracle Identity Analytics application resides on an application server and the Oracle Identity Analytics central repository of application data resides on a database server.

2.2 Installing and Readying Your Application Server

For a list of supported application servers, see the platform certification matrix.

2.2.1 Installing an Application Server

You can deploy Oracle Identity Analytics on any supported application server.

If deploying to Oracle WebLogic Server, you need to create a domain. Refer to the Oracle WebLogic Server documentation for details.

2.2.2 Configuring the Locale

The application server should be configured to use the same locale or encoding as the database and the Java Virtual Machine (JVM™ installation).

Inconsistent encodings can introduce certain globalization issues, such as the incorrect handling of multibyte characters. In globalized environments, UTF-8 should be implemented on all products.

Refer to your application server documentation for information about setting the locale and encoding.

2.2.3 Setting Up a Java Virtual Machine

A Java Virtual Machine (JVM) is required to run the Java classes that perform actions within Oracle Identity Analytics. To configure Oracle Identity Analytics for deployment you will need at least a version 5.0 JDK™. To run Oracle Identity Analytics on your application server at least a version 5.0 JDK™ or JRE™ is required. If integrating Oracle Identity Analytics and Oracle Identity Manager, both servers need to run the same version of the JDK or JRE.

Tip:

- Many application servers include JDK software bundled with their installation. The JDK version that is shipped with the application server is always preferred to any other JDK installed on your server.
- The JVM software should be configured to use the same locale or encoding as the application server and the database.

2.2.4 Configuring JVM Options

Refer to your application server's documentation for information about configuring JVM options.

- You should determine your memory need and set values in your application server's JVM accordingly. The recommended memory settings are as follows:

  ```
  -Xmx2048m -Xms2048m
  ```

  Note:

  - For a 32-bit system, a Java heap size exceeding `1536 MB` is not recommended.
  - Depending on your specific implementation, you might need to increase these recommended values if you face performance issues with the web interface.

- IPv4 is required by Oracle Identity Analytics for network communication. Add the following JVM option to enforce the IPv4 preference over IPv6:

  ```
  -Djava.net.preferIPv4Stack=true
  ```

  Also, ensure that your OS is configured to use IPv4. Refer to your operating system documentation for instructions about how to enable the IPv4 stack.

2.2.5 Configuring Environment Variables

Set the following environment variables on your application server(s) in preparation for deployment.

2.2.5.1 Configure the RBACX_HOME Environment Variable

Before deploying Oracle Identity Analytics you need to configure the `RBACX_HOME` environment variable. The `RBACX_HOME` variable should be permanently set to the

directory where you will deploy Oracle Identity Analytics. To create a permanent environment variable, refer to your operating system documentation for instructions.

Note:

- A permanent `$RBACX_HOME` environment variable should be created under the application server's owner profile. Oracle Identity Analytics deployment will fail if `RBACX_HOME` is declared in a profile inaccessible by the application server.
- For a clustered deployment, the `$RBACX_HOME` environment variable needs to be created on every cluster member.

2.2.5.2 Configure the Database Home Environment Variable

If your database is installed on the same machine on which you will deploy Oracle Identity Analytics, create an environment variable that maps to your database installation directory. For example, if you are using Oracle Database, create a permanent `$ORACLE_HOME` environment variable and map it to your Oracle installation directory.

3

Configuring Your Oracle Identity Analytics Installation Prior to Deployment

This chapter contains the following sections:

In this chapter you will you will create the Oracle Identity Analytics schema on the database server and you will customize the Oracle Identity Analytics WAR file before you deploy it on your application server.

You can perform the steps in this chapter either on the system on which OIA will be deployed or on a local system. If you configure OIA on a local system, your system should have the same installed OS as the system to which OIA will be deployed.

3.1 Create the RBACX_HOME Environment Variable on Your Local System

If the system that you are using to configure Oracle Identity Analytics is not the system on which you will deploy OIA, complete the steps in this section to create the RBACX_HOME environment variable. Otherwise, go to the next section.

> **Note:** To create a permanent environment variable on your application server system(s), see Section 2.2.5, "Configuring Environment Variables."

To Create the RBACX_HOME Environment Variable on Your Local System

Before configuring Oracle Identity Analytics you need to create the `RBACX_HOME` environment variable.

To temporarily set the $RBACX_HOME environment variable, type the following command at a command line. The $RBACX_HOME environment variable will remain valid for the length of your session.

- **Windows:**

  set RBACX_HOME=*Path to the Oracle Identity Analytics installation directory*

  For example: set RBACX_HOME=C:\Oracle\OIA_Install

- **UNIX:**

  export RBACX_HOME=*Path to the Oracle Identity Analytics installation directory*

  For example: export RBACX_HOME=/opt/Oracle/OIA_Install

3.2 Create the Oracle Identity Analytics Folder Structure

Before You Begin - Prior to installing Oracle Identity Analytics, verify that the following prerequisites have been met:

- You should have installed your database.
- If your database is installed on the same machine on which you are installing Oracle Identity Analytics, you should create an environment variable that maps to your database installation directory. For example, if you are using Oracle Database, create an $ORACLE_HOME environment variable and map it to your Oracle installation directory.
- If installing on a UNIX® based system, you should have sufficient privileges to create and modify folders in the /opt directory of the file system.
- To execute the database schema on the target database, DBA privileges are required. Use system for Oracle Database or an account with equivalent privileges.

1. At a command prompt, type the following commands to create the necessary folder structure.

   - **Windows:**

     ```
     C:\> mkdir Oracle\OIA_Install
     ```

   - **UNIX:**

     ```
     $ mkdir -p /opt/Oracle/OIA_Install
     ```

2. Unpack the Oracle Identity Analytics installation package.

   - **Windows:**

     unzip *oia_install_package.zip* -d OIA_Install

   - **UNIX:**

     unzip *oia_install_package.zip* -d OIA_Install

3. Verify that the Oracle Identity Analytics folder structure was properly created.

   The directory that you installed Oracle Identity Analytics to should consist of at least the following folders and files.

Table 3–1 Contents of the Base OIA Directory

Name	Type
.indexes	*folder*

Table 3–1 (Cont.) Contents of the Base OIA Directory

Name	Type
conf	*folder*
db	*folder*
legal	*folder*
rbacx.war	*file*
reports	*folder*
sample	*folder*

Note: The sample folder contains import and export folders that show the folder structure recommended for file and ETL imports.

4. Copy the import and export folders (located in the sample folder) to *$RBACX_HOME*. The import and export folders should be at the same level as conf, .indexes, and reports.

3.3 Download the Third-Party Library Files

Whereas some third-party files are available in the sample folder in the OIA package, other files need to be downloaded prior to installation. Third-party files should be saved to a library folder, which will be referred to as OIA_LIB.

For example:

- **Windows:**

  mkdir C:\Oracle\OIA_Lib

  set OIA_LIB=*Path to the downloaded third-party library files*
  (for example, set OIA_LIB=C:\Oracle\OIA_Lib)

- **UNIX**:

  mkdir /opt/Oracle/OIA_Lib

  export OIA_LIB=*Path to the downloaded third-party library files*
  (for example, export OIA_LIB=/opt/Oracle/OIA_Lib)

3.3.1 JDBC Drivers

For Oracle Database Server, download from the Oracle website the ojdbc5.jar driver if using JDK 1.5, and download the ojdbc6.jar driver if using JDK 1.6. The JDBC driver file you choose needs to support both the JDK version you are running, as well as the backend Oracle database instance version.

3.3.2 The jasper-jdt.jar File

This file is required by the Oracle Identity Analytics certification and reporting feature.

Download the jasper-jdt.jar file and paste it to the *OIA_LIB* folder:

http://tomcat.apache.org/dev/dist/m2-repository/org/apache/tomcat/jasper-jdt/6.0.18/

3.3.3 The CloverETL Library

OIA uses CloverETL for data import and export transformations.

Download the CloverETL Engine class files, version 1.8.1 from this site:

http://download.berlios.de/cloveretl/cloverETL.rel-1-8-1.zip

Next, complete the steps in the following section to convert the library file to a JAR file.

3.3.3.1 To Convert the Downloaded CloverETL Library File to a JAR File

Follow these steps to create the CloverETL JAR file.

Before You Begin - You will need the `unzip` utility and at least a Java 5 JDK.

1. Create a working directory named `files` and then open the directory:

   ```
   mkdir files
   ```

   ```
   cd files
   ```

2. Check the integrity of the `.zip` file and then expand it:

   ```
   unzip -tq ../cloverETL.rel-1-8-1.zip
   ```

   ```
   unzip -q ../cloverETL.rel-1-8-1.zip
   ```

3. Remove the `log4j.properties` file to prevent a file conflict in OIA:

   ```
   rm log4j.properties
   ```

4. Go to the parent directory:

   ```
   cd ..
   ```

5. Create the JAR manifest input file `clover.mf`, which consists of these lines:

   ```
   Implementation-Version: 1.8.1
   Implementation-Title: jETeL/Clover
   Implementation-URL:http://download.berlios.de/cloveretl/cloverETL.rel-1-8-1.zip
   Implementation-Vendor-Id: org.jetel
   ```

6. Create the JAR file:

   ```
   jar cfm clover.mf cloverETL-1.8.1.jar -C files .
   ```

7. Copy the `cloverETL-1.8.1.jar` file to the *OIA_LIB* folder.

3.3.4 The jxl-2.5.9.jar File

OIA uses the Java-Excel API to import data from an Excel spreadsheet file.

Download the `jxl-2.5.9.jar` file from this site and paste it to the *OIA_LIB* folder:

http://www.andykhan.com/jexcelapi/

3.3.5 The Web Services Description Language for Java Toolkit (WSDL4J)

OIA uses the Web Services Description Language for Java Toolkit (`WSDL4J.jar`) for provisioning server integration, among other things. If you are using Oracle Identity Analytics Web Services, download the WSDL4J JAR file.

3.3.5.1 To Download and Extract the WSDL4J JAR File

1. Download the `wsdl4j-bin-1.6.1.zip` file from this site:

http://sourceforge.net/projects/wsdl4j/files/WSDL4J/1.6.1/

Next, follow these steps to extract the `wsdl4j.jar` file and copy it to the `OIA_LIB` folder.

2. Verify the zip file was downloaded without errors:
   - **Windows:**

     ```
     unzip -tq wsdl4j-bin-1.6.1.zip
     ```

   - **UNIX:**

     ```
     unzip -tq wsdl4j-bin-1.6.1.zip
     ```

3. Extract the `wsdl4j.jar` file:
   - **Windows:**

     ```
     unzip -q wsdl4j-bin-1.6.1.zip wsdl4j-1_6_1\lib\wsdl4j.jar
     ```

   - **UNIX:**

     ```
     unzip -q wsdl4j-bin-1.6.1.zip wsdl4j-1_6_1/lib/wsdl4j.jar
     ```

4. Copy the JAR file to the `OIA_LIB` folder and rename it to include the version number:
   - **Windows:**

     ```
     move wsdl4j-1_6_1\lib\wsdl4j.jar %OIA_LIB%\wsdl4j-1.6.1.jar
     ```

   - **UNIX:**

     ```
     mv wsdl4j-1_6_1/lib/wsdl4j.jar $OIA_LIB/wsdl4j-1.6.1.jar
     ```

5. Clean up:
   - **Windows:**

     ```
     del /F wsdl4j-bin-1.6.1.zip  wsdl4j-1_6_1\
     ```

   - **UNIX:**

     ```
     rm -fr  wsdl4j-bin-1.6.1.zip  wsdl4j-1_6_1/
     ```

3.4 To Create the Oracle Identity Analytics Schema on the Database Server

Before You Begin - Prior to creating the Oracle Identity Analytics schema on the Database Server, review the following:

- You should have created the Oracle Identity Analytics folder structure as described in Section 3.2, "Create the Oracle Identity Analytics Folder Structure."
- If the database is installed on a remote machine, copy the `Oracle/OIA_Install/db` folder to the database machine.
- If it is necessary to use a database name other than `rbacx` and a user name other than `rbacxservice`, open the schema creation script and replace all instances of the default database name and default user name with the names that you will be using instead.

3.4.1 Oracle Database

In the following steps you will create the `rbacxservice` user, run the schema creation script, and then verify that the schema was created.

To Create the rbacxservice User and run the Schema Creation Script

1. Open a command prompt and type `oraenv`.
2. At the `ORACLE_SID` prompt, type `rbacx`.
3. Access the database using SQL*Plus:
   a. Type `sqlplus` at the command prompt.
   b. Type the user name and password for your database when prompted.
4. At the SQL prompt, type the following:

   `create user rbacxservice identified by` *your-database-password*`;`

   where *your-database-password* is the password for your database.

   For example:

   `create user rbacxservice identified by Welcome1;`
5. At the SQL prompt, type the following:
   - **Windows:**

     `@C:\Oracle\OIA_Install\db\oracle\rbacx-`*version*`_oracle_schema.sql`
   - **UNIX:**

     `@/opt/Oracle/OIA_Install/db/oracle/rbacx-`*version*`_oracle_schema.sql`

   Note: For Oracle Identity Analytics 11gR1 PS1 the version number is 11.1.1.5.0.

   The OIA schema creation script runs.

   When the script is finished, the SQL prompt appears.

To Verify That the Schema was Created

1. Open the Oracle Enterprise Manager 11g Database Control and type the user name and password for your database.

   Select **Normal** from the **Connect As** menu.
2. Choose **Schema** from the menu, then choose **Tables** from the **Database Objects** list.
3. Click the search icon to the right of the **Schema** field.
4. In the **Search And Select: Schema** window, type `RBACXSERVICE` in the **Schema** field and click **Go**.

   You should see one match in the search results.

   Note: To optimize performance, Oracle recommends that a qualified database administrator perform database tuning steps. These steps are included in the Database Tuning chapter in the *Database Administrator's Guide for Oracle Identity Analytics*.

3.5 To Configure Oracle Identity Analytics for Standalone Deployment

Follow these steps to build a custom `.war` file for your environment.

Before You Begin -

- An installed JDK is required (Version 1.5, at minimum).
- You should have completed the steps in Section 3.2, "Create the Oracle Identity Analytics Folder Structure."
- You should have saved the required third-party files to the `OIA_LIB` library file. See Section 3.3, "Download the Third-Party Library Files" for more information.
- You should have downloaded the JDBC connectivity JAR file for your database and saved it to the `OIA_LIB` library file. See Section 3.3.1, "JDBC Drivers" for more information.
- You should have created the Oracle Identity Analytics schema on the database server (see Section 3.4).

1. Create a backup copy of the original `rbacx.war` file.
   - **Windows:**

     ```
     C:\> cd C:\Oracle\OIA_Install
     C:\Oracle\OIA_Install> mkdir rbacx_original
     C:\Oracle\OIA_Install> copy rbacx.war rbacx_original
     ```

     A copy of the `rbacx.war` file is created under `C:\Oracle\OIA_Install\rbacx_original`.
   - **UNIX:**

     ```
     $ cd /opt/Oracle/OIA_Install
     $ mkdir rbacx_original
     $ cp rbacx.war rbacx_original/.
     ```

     A copy of the `rbacx.war` file is created under `/opt/Oracle/OIA_Install/rbacx_original`.
2. Create an `rbacx_staging` folder under `$RBACX_HOME`.
   - **Windows:**

     ```
     C:\Oracle\OIA_Install> mkdir rbacx_staging
     C:\Oracle\OIA_Install> cd rbacx_staging
     ```
   - **UNIX:**

     ```
     $ mkdir rbacx_staging
     $ cd rbacx_staging
     ```
3. Extract `rbacx.war` to `rbacx_staging` so that configuration changes can be made.
   - **Windows:**

     ```
     C:\Oracle\OIA_Install\rbacx_staging> jar -xvf ../rbacx.war
     ```
   - **UNIX:**

     ```
     $ jar xvf ../rbacx.war
     ```
4. Navigate to `rbacx_staging/WEB-INF`.

- **Windows:**

  ```
  C:\Oracle\OIA_Install\rbacx_staging> cd WEB-INF
  ```

- **UNIX:**

  ```
  $ cd WEB-INF
  ```

5. Update the `log4j.properties` file with the correct path for your environment.

   Note: If the Oracle Identity Analytics log file is going to be created in any folder other than the default log folder as defined by the application server, complete the step. Otherwise, skip and go to the next step.

   a. In a text editor, open the `log4j.properties` file located in the `WEB-INF` folder.

   b. Locate the following line under `# File Appender`.

   ```
   log4j.appender.file.file=logs/rbacx.log
   ```

   c. Replace `logs/rbacx.log` with the full path to where the log file should be written.

   For example, the line should look like this:

   – **Windows:**
   ```
   log4j.appender.file.file=C:/Oracle/OIA_Install/logs/rbacx.log
   ```

   or

   ```
   log4j.appender.file.file=E:\\Ora-
   cle\\OIA_Install\\logs\\rbacx.log
   ```

   – **UNIX:**
   ```
   log4j.appender.file.file=/opt/Oracle/OIA_Install/logs/rbacx.log
   ```

6. Update the `jasper.properties` file.

   a. In a text editor, open the `jasper.properties` file located in the `WEB-INF` folder.

   b. Add the following line to the end of the file:

   `net.sf.jasperreports.compiler.classpath=`*Path to your rbacx folder*`/rbacx/WEB-INF/lib/jasperreports-2.0.5-javaflow.jar`

   Note: The path to the `rbacx` deployment folder will vary on the application server.

7. If you are using CloverETL, enable it in the configuration as follows:

   a. In a text editor, open `WEB-INF/etl-context.xml` and uncomment the `etlManager` bean definition.

   b. In a text editor, open `WEB-INF/iam-context.xml` and uncomment the `etlManager` bean reference in the property list of the file bean definition.

8. Copy the downloaded third-party library files to the Oracle Identity Analytics library under the `WEB-INF/lib` directory.

   – **Windows:**

   ```
   C:\> cd C:\Oracle\OIA_Install\rbacx_staging
   ```

```
C:\Oracle\OIA_Install\rbacx_staging> copy %OIA_LIB% WEB-INF\lib
```

– **UNIX:**

```
$ cp $OIA_LIB/* WEB-INF/lib
```

9. Make the following changes if there are multiple instances of Oracle Identity Analytics, standalone or clustered, on the same subnet.
   a. Navigate to `rbacx_staging/WEB-INF` directory.
   b. In a text editor, open `application-context.xml`, find bean ID `commManager`, and examine the `constructor-arg` value.
   c. Set the `constructor-arg` value with a unique instance name—for example, `value="OIA-Instance-1"`.
   d. In a text editor, open `search-context.xml`, find bean ID `searchConfiguration`, and examine the `constructor-arg` value.

      If the deployment is standalone, `constructor-arg` defaults to a value of `0`, which is specified as `value="0"`.
10. If multiple instances of Oracle Identity Analytics, standalone or clustered, exist on the same subnet, navigate to `rbacx_staging/WEB-INF/classes` directory and do the following:
    a. In a text editor, open `oscache.properties` (located in the `rbacx_staging/WEB-INF/classes` directory), and find the `cache.cluster.multicast.ip` property.
    b. Uncomment `cache.cluster.multicast.ip` by removing the `#` at the start of the line. Each Oracle Identity Analytics instance requires a unique `cache.cluster.multicast.ip` value.
    c. Uncomment the following line by removing the # at the start of the line.

       ```
       cache.event.listeners=com.opensymphony.oscache.plugins.clustersupport.JavaG
       roupsBroadcastingListener,com.opensymphony.oscache.extra.CacheMapAccessEven
       tListenerImpl
       ```
11. If you are using OIA Web Services, uncomment its configuration.

    See the *API Guide for Oracle Identity Analytics*, "Enabling Web Services."
12. Run the following commands from the `rbacx_staging` folder to repack the newly modified `rbacx.war` file.

    The expanded folder structure is repackaged to a `.war` file, which is required for deployment on the application server.

    - **Windows:**

      ```
      C:\> cd C:\Oracle\OIA_Install\rbacx_staging
      ```

      ```
      C:\Oracle\OIA_Install\rbacx_staging> jar -cvfM ../rbacx.war .
      ```

      The new `rbacx.war` file is located in `C:\Oracle\OIA_Install`.

    - **UNIX:**

      ```
      $ cd /opt/Oracle/OIA_Install/rbacx_staging
      ```

      ```
      $ jar cvfM ../rbacx.war .
      ```

      The new `rbacx.war` file is located in `/opt/Oracle/OIA_Install`.

13. Copy the `jdbc.properties` file for your database to the Oracle Identity Analytics `conf` directory, located as follows.

    - **Windows:**

      ```
      cd C:\Oracle\OIA_Install\conf

      copy oracle\jdbc.properties .
      ```

    - **UNIX:**

      ```
      cd /opt/Oracle/OIA_Install/conf

      cp oracle/jdbc.properties .
      ```

14. In a text editor, open the `jdbc.properties` file and edit the following lines, substituting *$SERVER_NAME* and *$PORT_NUMBER* with the host name and connectivity port of the target database.

    For Oracle Database:

    ```
    # JDBC driver URL

    jdbc.url=jdbc:oracle:thin:@$SERVER_NAME:$PORT_NUMBER:rbacx
    ```

    The default port number for JDBC connectivity on the Oracle server is `1521` and it is assumed that the default SID is `rbacx`.

15. If you are using a database user name other than the default `rbacxservice` user name, change the `jdbc.username` value in the `jdbc.properties` file to the user name that you created. Otherwise, go to the next step.

16. In the `jdbc.properties` file, add the following line:

    `jdbc.password=`*YourClearTextDatabasePassword*

    Now encrypt the database password by typing the following command at a command line:

    - **Windows:**

      ```
      C:\> java -jar ..\rbacx_staging\WEB-INF\lib\vaau-commons-crypt.jar
      -encryptProperty -cipherKeyProperties .\cipherKey.properties
      -propertyFile .\jdbc.properties -propertyName jdbc.password
      ```

    - **UNIX:**

      ```
      $ java -jar ../rbacx_staging/WEB-INF/lib/vaau-commons-crypt.jar
      -encryptProperty -cipherKeyProperties ./cipherKey.properties
      -propertyFile ./jdbc.properties -propertyName jdbc.password
      ```

    For more information about this command, see "Understanding the Property Encryption Utility" in the "Securing Oracle Identity Analytics" chapter of the *Administrator's Guide for Oracle Identity Analytics*.

    The password is encrypted and stored as `jdbc.password.encrypted`.

17. Delete the `rbacx_staging` folder.

    - **Windows:**

      ```
      C:\> cd C:\Oracle\OIA_Install

      C:\> rmdir /s rbacx_staging
      ```

    - **UNIX:**

      ```
      $ cd /opt/Oracle/OIA_Install
      ```

```
$ rm -rf rbacx_staging
```

18. Do the following:

    a. Go to the following location and open `iam.properties` in a text editor.

       – **Windows:**

         `C:\>Oracle\OIA_Install\conf`

       – **UNIX:**

         `/opt/Oracle/OIA_Install/conf`

    b. Edit the following lines, replacing `$RBACX_HOME` with the path to the directory where Oracle Identity Analytics is installed:

```
com.vaau.rbacx.iam.file.import.completeLocation=$RBACX_HOME/import/complete
com.vaau.rbacx.iam.file.import.schemaLocation=$RBACX_HOME/import/schema
com.vaau.rbacx.iam.file.import.dropLocation=$RBACX_HOME/import/in
com.vaau.rbacx.etl.import.dropLocation=$RBACX_HOME/import/etl/in
com.vaau.rbacx.etl.import.graphsLocation=$RBACX_HOME/import/etl/graphs
com.vaau.rbacx.etl.import.completeLocation=$RBACX_HOME/import/etl/complete
com.vaau.rbacx.etl.import.outputLocation=$RBACX_HOME/import/in
com.vaau.rbacx.iam.file.export.dropLocation=$RBACX_HOME/export/etl/in
com.vaau.rbacx.iam.file.export.schemaLocation=$RBACX_HOME/export/schema
com.vaau.rbacx.etl.export.dropLocation=$RBACX_HOME/export/etl/in
com.vaau.rbacx.etl.export.graphsLocation=$RBACX_HOME/export/etl/graphs
com.vaau.rbacx.etl.export.completeLocation=$RBACX_HOME/export/etl/complete
com.vaau.rbacx.etl.export.outputLocation=$RBACX_HOME/export/out
```

3.6 To Configure Oracle Identity Analytics for Clustered Deployment

Follow these steps to build a custom `.war` file for your environment.

> **Note:** Before deploying Oracle Identity Analytics to a cluster, synchronize the clocks on every machine in the cluster. The clocks must be within one second of each other.

Before You Begin -

- Complete the steps in Section 3.2, "Create the Oracle Identity Analytics Folder Structure."
- An installed JDK is required (Version 1.5, at minimum).
- You should have downloaded the JDBC connectivity JAR file for your database. See Section 3.3.1, "JDBC Drivers" for more information.
- You should have created the Oracle Identity Analytics schema on the database server (see Section 3.4).

1. Create a backup copy of the original `rbacx.war` file.

   - **Windows:**

```
C:\> cd C:\Oracle\OIA_Install

C:\Oracle\OIA_Install> mkdir rbacx_original

C:\Oracle\OIA_Install> copy rbacx.war rbacx_original
```

A copy of the `rbacx.war` file is created under `C:\Oracle\OIA_Install\rbacx_original`

- **UNIX:**

```
$ cd /opt/Oracle/OIA_Install
$ mkdir rbacx_original
$ cp rbacx.war rbacx_original/.
```

A copy of the `rbacx.war` file is created under `/opt/Oracle/OIA_Install/rbacx_original`

2. Create an `rbacx_staging` folder under `$RBACX_HOME`.

- **Windows:**

```
C:\Oracle\OIA_Install> mkdir rbacx_staging
C:\Oracle\OIA_Install> cd rbacx_staging
```

- **UNIX:**

```
$ mkdir rbacx_staging
$ cd rbacx_staging
```

3. Extract `rbacx.war` to `rbacx_staging` so that configuration changes can be made.

- **Windows:**

```
C:\Oracle\OIA_Install\rbacx_staging> jar -xvf ../rbacx.war
```

- **UNIX:**

```
$ jar xvf ../rbacx.war
```

4. Navigate to `rbacx_staging/WEB-INF`.

- **Windows:**

```
C:\Oracle\OIA_Install\rbacx_staging> cd WEB-INF
```

- **UNIX:**

```
$ cd WEB-INF
```

5. Update the `log4j.properties` file with the correct path for your environment.

Note: If the Oracle Identity Analytics log file is going to be created in any folder other than the default log folder as defined by the application server, complete the step. Otherwise, skip and go to the next step.

---

a. In a text editor, open the `log4j.properties` file located in the `WEB-INF` folder.

b. Locate the following line under `# File Appender`.

```
log4j.appender.file.file=logs/rbacx.log
```

c. Replace `logs/rbacx.log` with the full path to where the log file should be written.

For example, the line should look like this:

– **Windows:**
`log4j.appender.file.file=C:/Oracle/OIA_Install/logs/rbacx.log`

- **UNIX:**
  `log4j.appender.file.file=/opt/Oracle/OIA_Install/logs/rbacx.log`

6. Update the `jasper.properties` file.
   a. In a text editor, open the `jasper.properties` file located in the `WEB-INF/classes` folder.
   b. Add the following line to the end of the file:

      `net.sf.jasperreports.compiler.classpath=`*Path to your rbacx folder*`/rbacx/WEB-INF/lib/jasperreports-2.0.5-javaflow.jar`

      Note: The path to the `rbacx` deployment folder will vary on the application server.

7. If you are using CloverETL, enable it in the configuration as follows:
   a. In a text editor, open `WEB-INF/etl-context.xml` and uncomment the `etlManager` bean definition.
   b. In a text editor, open `WEB-INF/iam-context.xml` and uncomment the `etlManager` bean reference in the property list of the file bean definition.
8. Copy the downloaded third-party library files to the Oracle Identity Analytics library under the `WEB-INF/lib` directory.

   Note: Ensure that the oscache.jar and javagroups-all.jar JAR files are not part of the third-party library files being copied in this step. If they are, do not copy the JAR files to the `WEB-INF/lib` directory of the Oracle Identity Analytics library.

   If these JAR files are copied, the OIA managed servers in the cluster will fail to start the OIA application.

   - **Windows:**

     ```
     copy %OIA_LIB% * WEB-INF\lib
     ```

   - **UNIX:**

     ```
     cp $OIA_LIB/ * WEB-INF/lib
     ```

9. Make the following changes to enable Oracle Identity Analytics 11gR1 support for clustered application server deployments.
   a. Navigate to the `rbacx_staging/WEB-INF` directory.
   b. In a text editor, open `application-context.xml`, find bean ID `commManager`, and examine the `constructor-arg` value.
   c. Set the `constructor-arg` value as the cluster name—for example, `value="Prod-1-Cluster"`.
      – If Oracle Identity Analytics is deployed on multiple clusters within the same subnet, you should define unique `constructor-arg` values for each deployment. For example, if both clusters Prod-Cluster and QA-Cluster have Oracle Identity Analytics deployed, the `constructor-arg` values of each should be set to `Prod-Cluster` and `QA-Cluster` respectively.

- Members of the same cluster should have the same `constructor-arg` value.

d. In bean ID `commManager`, locate the `constructor-arg index="1"` value.

e. Replace the value with the IP address of each cluster member. This setting binds the multicast addresses to the IP addresses. In addition, add the `enabled` property and set it to `true`. For example:

```
<constructor-arg index="1" value="140.84.134.133;140.84.135.88"/>
<property name="enabled" value="true"/>
```

Save the `application-context.xml` file.

f. In a text editor, open `search-context.xml`, find bean ID `searchConfiguration`, and examine the `constructor-arg` value.

- If the deployment is a clustered deployment, `constructor-arg` defaults to a value of `1` or `2` depending on the location of the `.indexes` directory.

To set the `constructor-arg` value, do the following:

- If each clustered node will be accessing *local* individual `.indexes` directories, set `constructor-arg` to `1`. For example, `value="1"`.
- If clustered nodes will be accessing *a shared* `.indexes` directory, set `constructor-arg` to `2`. For example, `value="2"`. The `.indexes` directory needs to be located on an NFS share location where each clustered node has read-write permission. Edit `indexLocation` such that the NFS share location replaces `$RBACX_HOME` in the `value` field.

  If clustered nodes will be accessing *a shared* `.indexes` directory, then copy the $RBACX_HOME/.indexes directory structure across the shared indexes location. Perform the following procedure to do so:

To copy the directory structure from $RBACX_HOME/.indexes to the shared indexes location, do the following:

a. Undeploy the OIA from weblogic cluster.

b. Re-copy the ./indexes from the OIA installation media.

c. In the RBACX_HOME/rbacx.war/WEB-INF/lib directory, remove the stax-api-1.0.1.jar file or rename it to stax-api-1.0.1.jar.backup on the 2 servers.

d. Download stax-api-1.0-2.jar from `http://download.java.net/maven/1/javax.xml.stream/jars/` and place it in the RBACX_HOME/rbacx.war/WEB-INF/lib directory on the 2 servers.

e. Redeploy RBACX_HOME/rbacx.war from the Weblogic console.

g. If multiple instances of Oracle Identity Analytics, standalone or clustered, exist on the same subnet, navigate to the `rbacx_staging/WEB-INF/classes` directory and do the following:

a. In a text editor, open `oscache.properties` (located in the `rbacx_staging/WEB-INF/classes` directory), and find the `cache.cluster.multicast.ip` property.

b. Uncomment `cache.cluster.multicast.ip` by removing the # at the start of the line. Each non-member instance requires a unique `cache.cluster.multicast.ip` value.

c. Uncomment the following line by removing the # at the start of the line.

```
cache.event.listeners=com.opensymphony.oscache.plugins.clustersupport.J
avaGroupsBroadcastingListener,com.opensymphony.oscache.extra.CacheMapAc
cessEventListenerImpl
```

10. If you are using OIA Web Services, uncomment its configuration.

    See the *API Guide for Oracle Identity Analytics*, "Enabling Web Services."

11. Run the following commands from the `rbacx_staging` folder to repack the newly modified `rbacx.war` file. The expanded folder structure is repackaged to a `.war` file, which is required for deployment on the application server.

    - **Windows:**

      ```
      C:\> cd C:\Oracle\OIA_Install\rbacx_staging
      C:\> jar -cvfM ../rbacx.war .
      ```

      The new `rbacx.war` file is located in `C:\Oracle\OIA_Install`.

    - **UNIX:**

      ```
      $ cd /opt/Oracle/OIA_Install/rbacx_staging
      $ jar cvfM ../rbacx.war .
      ```

      The new `rbacx.war` file is located in `/opt/Oracle/OIA_Install`.

12. Copy the `jdbc.properties` file for your database to the Oracle Identity Analytics conf directory, located as follows.

    - **Windows:**

      ```
      cd C:\Oracle\OIA_Install\conf
      copy oracle\jdbc.properties .
      ```

    - **UNIX:**

      ```
      cd /opt/Oracle/OIA_Install/conf
      cp oracle/jdbc.properties .
      ```

13. In a text editor, open the `jdbc.properties` file and edit the following lines, substituting *$SERVER_NAME* and *$PORT_NUMBER* with the host name and connectivity port of the target database.

    For Oracle Database:

    ```
    # JDBC driver URL
    jdbc.url=jdbc:oracle:thin:@$SERVER_NAME:$PORT_NUMBER:rbacx
    ```

    The default port number for JDBC connectivity on the Oracle server is `1521` and it is assumed that the default SID is `rbacx`.

14. If you are using a database user name other than the default `rbacxservice` user name, change the `jdbc.username` value in the `jdbc.properties` file to the user name that you created. Otherwise, go to the next step.

15. Make the following change to `jdbc.properties` for clustered Quartz support, and save the file:

    ```
    jdbc.quartz.isClustered=true
    ```

16. In the `jdbc.properties` file, add the following line:

    ```
    jdbc.password=YourClearTextDatabasePassword
    ```

Now encrypt the database password by typing the following command at a command line:

- **Windows:**

```
C:\> java -jar ..\rbacx_staging\WEB-INF\lib\vaau-commons-crypt.jar
-encryptProperty -cipherKeyProperties .\cipherKey.properties
-propertyFile .\jdbc.properties -propertyName jdbc.password
```

- **UNIX:**

```
$ java -jar ../rbacx_staging/WEB-INF/lib/vaau-commons-crypt.jar
-encryptProperty -cipherKeyProperties ./cipherKey.properties
-propertyFile ./jdbc.properties -propertyName jdbc.password
```

For more information about this command, see "Understanding the Property Encryption Utility" in the "Securing Oracle Identity Analytics" chapter of the *Administrator's Guide for Oracle Identity Analytics*.

The password is encrypted and stored as `jdbc.password.encrypted`.

Repeat steps 12 – 16 for each additional OIAinstance, or, if every OIA instance uses identical JDBC connection properties, copy the `jdbc.properties` file to all instances.

17. Delete the `rbacx_staging` folder.

    - **Windows:**

      ```
      C:\> cd C:\Oracle\OIA_Install

      C:\> rmdir /s rbacx_staging
      ```

    - **UNIX:**

      ```
      $ cd /opt/Oracle/OIA_Install

      $ rm -rf rbacx_staging
      ```

18. Do the following:

    a. Go to the following location and open `iam.properties` in a text editor.

       – **Windows:**

         ```
         C:\>Oracle\OIA_Install\conf
         ```

       – **UNIX:**

         ```
         /opt/Oracle/OIA_Install/conf
         ```

    b. Edit the following lines, replacing *$RBACX_HOME* with the path to the directory where Oracle Identity Analytics is installed:

       ```
       com.vaau.rbacx.iam.file.import.completeLocation=$RBACX_HOME/import/complete
       com.vaau.rbacx.iam.file.import.schemaLocation=$RBACX_HOME/import/schema
       com.vaau.rbacx.iam.file.import.dropLocation=$RBACX_HOME/import/in
       com.vaau.rbacx.etl.import.dropLocation=$RBACX_HOME/import/etl/in
       com.vaau.rbacx.etl.import.graphsLocation=$RBACX_HOME/import/etl/graphs
       com.vaau.rbacx.etl.import.completeLocation=$RBACX_HOME/import/etl/complete
       com.vaau.rbacx.etl.import.outputLocation=$RBACX_HOME/import/in
       com.vaau.rbacx.iam.file.export.dropLocation=$RBACX_HOME/export/etl/in
       com.vaau.rbacx.iam.file.export.schemaLocation=$RBACX_HOME/export/schema
       com.vaau.rbacx.etl.export.dropLocation=$RBACX_HOME/export/etl/in
       com.vaau.rbacx.etl.export.graphsLocation=$RBACX_HOME/export/etl/graphs
       com.vaau.rbacx.etl.export.completeLocation=$RBACX_HOME/export/etl/complete
       com.vaau.rbacx.etl.export.outputLocation=$RBACX_HOME/export/out
       ```

Note: When the application server is clustered, the nodes can maintain localized import/export directories, or utilize import/export directories on a NFS share. If the nodes use a shared NFS location for import/export, substitute `$RBACX_HOME` with the path to the NFS share.

3.7 To Configure J2EE Data Sources for Oracle Identity Analytics (Optional)

The following steps describe how to configure Oracle Identity Analytics to use a JDBC connection pool. The use of JDBC connection pooling with Oracle Identity Analytics is optional.

1. On your application server, create the data source and configure the JDBC connection. Refer to your application server documentation for instructions.

   Complete the following steps to point Oracle Identity Analytics to the data source.

2. Comment out the `jdbc.properties` configuration value using these steps:

   a. Open `conf-context.xml` (located in the `WEB-INF` directory inside the `WAR` file) for editing.

   b. Locate `<property name="locations">` and comment out the following value:

   ```
   <value>file:$RBACX_HOME/conf/jdbc.properties</value>
   ```

   For example:

   ```
   <property name="locations">

   <!-- <value>file:$RBACX_HOME/conf/jdbc.properties</value> -->
   ```

3. Open `dataaccess-context.xml` (located in the `WEB-INF` directory inside the `WAR` file) for editing.

   a. Comment out the bean with `id="dataSource"` and add the following bean below it:

   ```
   <bean id="dataSource"
   class="org.springframework.jndi.JndiObjectFactoryBean">
       <property name="jndiName" value="jdbc/RMConfig" />
   ```

   Replace the value `jdbc/RMConfig` with a JNDI reference to the JDBC connection that you created on your application server.

   For example:

   ```
   <!-- <bean id="dataSource" parent="abstractDataSource">
   <description>Default datasource that uses Oracle UCP as a pool
   implementation</description>
   <property name="connectionFactoryClassName"
   value="${jdbc.driverClassName}"/>
   <property name="URL" value="${jdbc.url}"/>
   <property name="user" value="${jdbc.username}"/>
   <property name="password" value="${jdbc.password.encrypted}"/>
   <property name="connectionPoolName" value="CONN_POOL"/>
   <property name="minPoolSize" value="5"/>
   <property name="maxPoolSize" value="150"/>
   <property name="initialPoolSize" value="5"/>
   <property name="inactiveConnectionTimeout" value="120"/>
   <property name="validateConnectionOnBorrow" value="true"/>
   ```

```
<property name="SQLForValidateConnection" value="select 1 from
globalusers"/>
<property name="maxStatements" value="10"/>
<property name="connectionFactoryProperties">
<bean
class="org.springframework.beans.factory.config.PropertiesFactoryBean">
<property name="location"
value="file:${RBACX_HOME}/conf/jdbcConnectionFactory.properties"/>
<property name="ignoreResourceNotFound" value="true"/>
</bean>
</property>
</bean> -->

   <bean id="dataSource"
class="org.springframework.jndi.JndiObjectFactoryBean">
    <property name="jndiName" value="jdbc/rbacx" />
```

4. Open `scheduling-context.xml` (located in the `WEB-INF` directory inside the WAR file) for editing:
   a. Locate the bean with `id="quartzSchedulerFactoryBean"`.
   b. Locate the property name `<property name="quartzProperties">`.
   c. Change the `${jdbc.quartz.driverDelegateClass}` key value to `org.quartz.impl.jdbcjobstore.oracle.OracleDelegate`.
   d. Change the `${jdbc.quartz.selectWithLockSQL}` key value to `SELECT * FROM {0}LOCKS WHERE LOCK_NAME = ? FOR UPDATE`.
   e. If utilizing a clustered configuration, change the `${jdbc.quartz.isClustered}` key value to `true`.
5. If utilizing a clustered configuration, open the `application-context.xml` file (located in the `WEB-INF` directory inside the WAR file) for editing:
   a. Locate the property name `<property name="quartzProperties">`.
   b. Change the `${jdbc.quartz.isClustered}` key value to `true`.

4

Deploying Oracle Identity Analytics

This chapter contains the following sections:

- Section 4.1, "Deploying on Tomcat"
- Section 4.2, "Deploying on WebSphere"
- Section 4.3, "Deploying on WebLogic"

Follow the steps on this page to deploy Oracle Identity Analytics to your application server.

Before You Begin-

- You should have created the OIA schema on the database server and configured the `rbacx.war` to deploy in your environment. For details, see Chapter 3, "Configuring Your Oracle Identity Analytics Installation Prior to Deployment."
- Verify that your JVM Maximum and Minimum Heap Size configuration meets the minimum values recommended for Oracle Identity Analytics. See Section 2.2.4, "Configuring JVM Options" for further information about these values.
- These instructions are for a stand-alone (non-clustered) Oracle Identity Analytics deployment.

> **Note:** When running two or more applications in the same administrative domain (or cell), configure the application server so that the application class-loader takes precedence over the parent class-loader.

4.1 Deploying on Tomcat

Follow the procedure in this section to deploy Oracle Identity Analytics on Tomcat. Before you begin, you should have already completed the procedures in Chapter 3, "Configuring Your Oracle Identity Analytics Installation Prior to Deployment."

4.1.1 To Deploy Oracle Identity Analytics on Tomcat

1. Stop the Tomcat application server.
2. Browse to the `tomcat install` directory.
3. Copy the configured WAR file to the `webapps` directory that is located in the `tomcat install` directory.
4. Start the Tomcat application server.

4.2 Deploying on WebSphere

Follow the procedure in this section to deploy Oracle Identity Analytics on WebSphere 7. Before you begin, you should have already completed the procedures in Chapter 3, "Configuring Your Oracle Identity Analytics Installation Prior to Deployment."

4.2.1 To Configure WebSphere 7 to Run Oracle Identity Analytics

1. In a browser, open the WebSphere Administrative Console:

   `http://`*Hostname*`:`*Port-Number*`/ibm/console/`

2. Choose **Servers > Server Types > WebSphere Application Servers**.
3. Choose your ***Server Name*** **> Java and Process Management > Process Definition > Java Virtual Machine**.
4. Specify the following value under **Generic JVM Requirements**, and save your changes.

   `-Xverify:none`

5. Choose **Servers > Server Types > WebSphere Application Servers**.
6. Choose your *Server Name* > **Web Container Settings > Web Container > Custom Properties**.
7. Create a new custom property using the following parameters and save your changes.

   Name: `com.ibm.ws.webcontainer.invokefilterscompatibility`

   Value: `true`

8. Restart the application server.

4.2.2 To Deploy Oracle Identity Analytics on WebSphere 7

1. In a browser, log in to the WebSphere Administrative Console.

   `http://`*Hostname*`:`*Port-Number*`/ibm/console/`

2. Choose **Application > New Application** in the left panel.
   a. Click **New Enterprise Application**.
   b. Specify the path to the `rbacx.war` file and click **Next**.
      – If the `rbacx.war` file is located on the local system, type its complete path under Local File System.
      – If the file is on a remote machine, specify the path for the remote file system.
   c. Select **Detailed - Show me all installation options and parameters**.
   d. On the **Choose to generate default bindings and mappings** page keep the default configuration and click **Next**.
3. Click **Continue** on the Application Security Warning page.

   The "Select Installation Options" page opens.

4. Complete the form:

a. To install the application to a location other than the default location, type the path in the **Directory to Install Application** field. For example, on UNIX systems:

   `/opt/IBM/Websphere/AppServer/installed Apps/Hostname`

b. Verify that the following options are selected:

   – Precompile JavaServer Pages files

   – Distribute application

   – User binary configuration

c. Enter the name of the application in the Application Name field, and click **Next**.

   The default application name is set to **rbacx_war**.

5. On the page "Map modules to servers," verify that the Oracle Identity Analytics application mapping is to the appropriate cluster/server and click **Next**.

   The "Provide options to compile JSPs" page opens.

6. Make the following changes and click **Next**:

   a. Select Web Module **Oracle Identity Analytics**.

   b. Change JDK Source Level to 15.

7. On the **Provide JSP reloading options for Web Modules** page keep the default configuration.

8. On the **Map shared libraries** page, verify your settings and click **Next**.

   On the **Map shared library relationships** page, verify your settings and click **Next**.

   On the **Map virtual hosts for Web modules** page, verify that the Oracle Identity Analytics application mapping is set to the appropriate virtual host and click **Next**.

9. On the **Map context roots for Web modules** page, provide context root `/rbacx` and click **Next**.

10. Choose **Applications > Application Types > WebSphere Enterprise Applications > rbacx > Manage Modules**, and do the following:

    a. Verify that the cluster/server mapping is correct.

    b. Select **Oracle Identity Analytics > Class loader order**.

    c. Under Class loader order, select **Class loaded with local class loader first (parent last)**, and save your changes.

    Note: When running two or more applications in the same cell, it is important that you configure the application server such that the application class loader can override the parent and provide its own version of a class. For WebSphere, choose **Class loaded with local class loader first (parent last)**, as documented above.

11. In a text editor, open the `solr.xml` file (`RBACX_HOME/.indexes/solr.xml`) and add `persistent="false"` in `com.vaau.commons.search.searchengine.solr.templates.solr.xml`.

12. Restart WebSphere.

Oracle Identity Analytics deployment on WebSphere is complete.

Note: If the system is not writing messages to the System Log file when Oracle Identity Analytics is deployed on WebSphere, try the following:

1. Open the following file in a text editor:

   `$RBACX_WAR/META-INF/services/org.apache.commons.logging.LogFactory`

2. Replace the existing value with the following value and save your changes:

   `org.apache.commons.logging.impl.Log4jFactory`

4.3 Deploying on WebLogic

Follow the procedure in this section to deploy Oracle Identity Analytics on WebLogic. Before you begin, you should have already completed the procedures in Chapter 3, "Configuring Your Oracle Identity Analytics Installation Prior to Deployment."

4.3.1 To Deploy Oracle Identity Analytics on WebLogic

To deploy Oracle Identity Analytics on WebLogic, you must first extract the `.war` file to a staging folder.

1. At the command line, navigate to *$RBACX_HOME*.

   Note: The *$RBACX_HOME* environment variable denotes the path to the directory to which you installed Oracle Identity Analytics.

   ---

2. Type the following commands at a command prompt to create a directory where the `.war` file should be exploded.

   These steps require a JDK.

   - **Windows:**

     ```
     C:\> mkdir rbacx

     C:\> cd rbacx

     C:\> jar -xvf ../rbacx.war
     ```

   - **UNIX:**

     ```
     $ mkdir rbacx

     $ cd rbacx

     $ jar xvf ../rbacx.war
     ```

3. In a browser log in to the WebLogic administrative console:

   `http://`*Hostname*`:`*Port-Number*`/console/login/LoginForm.jsp`

4. In the administration console, go to the left panel and click **Lock & Edit**, located under **Change Center**.

5. Click **Deployments**, located under **Domain Structure**.

6. Click **Install**, located in the main panel under **Deployments**.

The Install Application Assistant opens.

7. On the Locate Deployment To Install And Prepare For Deployment page, navigate to the `rbacx` directory created in step two and select the folder such that it lists the contents of `rbacx.war`.
8. Select the **rbacx_staging** directory and click **Next**.
9. Select the **Install This Deployment As An Application** option and click **Next**.
10. On the Optional Settings page, do the following:
    a. Ensure that the deployment is named **rbacx**.
    b. Under **Security**, select **DDOnly: Use Only Roles And Policies That Are Defined In This Deployment**.
    c. Under **Source Accessibility**, select **Use The Defaults Defined By The Deployment's Targets**.
    d. Click **Next**.
11. On the Review Your Choices And Click Finish page, click **Yes, Take Me to the Deployments Configuration Screen**.

    Click **Finish**.
12. On the Settings for rbacx page, click **Save**.
13. Click **Deployments** in the left panel.

    The Summary of Deployments panel opens.

A status of **Active** indicates that Oracle Identity Analytics has been successfully deployed.

Note: When running two or more applications in the same application server administrative domain, configure the application server to allow the application to pick the appropriate JAR files from the application's libraries.

For WebLogic set the `prefer-application-packages` element in `weblogic.xml` as follows.

```
<weblogic-web-app
xmlns="http://xmlns.oracle.com/weblogic/weblogic-web-app">
  <container-descriptor>
     <prefer-application-packages>
         <package-name>javax.wsdl.*</package-name>
         <!-- List the other package names from the library here-->
     </prefer-application-packages>
  </container-descriptor>
</weblogic-web-app>
```

Refer to your WebLogic documentation for further information.

---

5 Verifying the Installation

This chapter contains the following section:

5.1 Verifying That the Oracle Identity Analytics Installation was Successful

Use the following procedure to verify that your Oracle Identity Analytics installation was successful and that the application is working properly.

5.1.1 To Verify an Oracle Identity Analytics Installation

1. Browse to your application server log files and verify that the `rbacx.log` file is present. This file is created when Oracle Identity Analytics is deployed to the application server.
2. Open `rbacx.log` in a text editor and check for a message indicating that Oracle Identity Analytics started successfully.
3. Using a browser, open the following URL: `http://`*Hostname:Port-Number*`/rbacx`

   Note: Refer to your application server documentation to determine the port number your application server uses.

4. Use the following credentials to log in to the server:

   Username: `rbacxadmin`

   Password: `password`

   Upon logging in, the system automatically expires the default password for the rbacxadmin account and the Change Password page opens.
5. Create a new rbacxadmin password.

Part III

Upgrading Oracle Identity Analytics

Part III provides detailed information and instructions to help you upgrade your Oracle Identity Analytics installation.

Part III contains the following chapters:

6

Overview of the Upgrade Process

This chapter contains the following sections:

6.1 Why Upgrade?

There are several reasons to upgrade to the latest version of Oracle Identity Analytics:

1. Access to the latest advanced features and functionality
2. Access to the latest security enhancements
3. Continued eligibility for full support and services

6.2 Phases of the Upgrade Process

The upgrade process is divided into three major phases:

- **Phase 1 - Prepare to upgrade Oracle Identity Analytics**

  In phase one you document your existing environment, choose an upgrade version, and document any custom components.

- **Phase 2 - Test the upgrade**

  In phase two you create a test environment, download third party library files, and deploy the upgrade in a test environment.

- **Phase 3 - Upgrade Oracle Identity Analytics in the production environment**

  In phase three you create a production rollout plan and deploy the upgrade in the production environment.

6.3 A Recommended Task List for Upgrading

1. Read the release notes of both the version you are upgrading *from*, and the version you are upgrading *to*, in order to understand important product changes.
2. Document your existing environment, which includes the following:
   - Platform - Be sure to document your application server and database server configuration.

- Oracle Identity Analytics Installation - Document file folder locations and configuration settings.
- Custom Components - Document custom configurations and utilities.
- Custom Workflows - Document custom workflow approval steps.
- Custom Reports - Create a backup of any custom reports. Custom reports include changes made to default Oracle Identity Analytics reports and any other reports uploaded into the application.

3. Create a backup of the existing database and Oracle Identity Analytics installation directory (including the contents). For example, back up the `.indexes`, `conf`, `import`, `export`, and `reports` directories, the existing `.war` file, and the exploded `.war` directory within the application server. This gives you the ability to reinstate the working environment back to the previous release, if necessary.
4. Create a test environment that mirrors the functionality of the production environment.
5. Develop a comprehensive test plan. The test plan's objective is to ensure that all currently utilized product functionality remains operational after the test and before deployment in the production environment.
6. Check the version level of the infrastructure components, for example the operating system, the JDK software, the application server, the database, all resources, and any provisioning systems. Confirm that support for your current environment is still provided in the latest version of Oracle Identity Analytics.
7. Rebuild any custom Java classes and utilities against the target product libraries.
8. Complete the entire upgrade before attempting to start the application server and resuming activity.

In addition, consider the following:

1. Upgrade the database schema if this step is required in the current release. The necessary files are provided in the distribution package inside the `db` folder.
2. If OIA is integrated with Oracle Waveset, and if the `rm_idm_init.xml` file (located in the *$RBACX_HOME*`/conf` directory) has been modified for a customized Oracle Waveset integration, extract and customize the file again to reflect the previous customizations.
3. If the `rbacxmessages.properties` file (located in the `WEB-INF/classes` directory) is modified for customized messages, extract and customize the file again to reflect the previous customizations.
4. If the `rbacxaudit-messages.properties` file (located in the `WEB-INF/classes` directory) is modified for customized messages, extract and customize the file again to reflect the previous customizations.
5. If a customized logo is required, create a backup of the `logo.gif` file located here:

   exploded_rbacx.war_directory`/images`
6. Back up any customized report(s) located in *OIA_Installation_Directory*`/reports`.
7. If the `jobs.xml` file was previously extracted from the `exploded_rbacx.war_directory/WEB-INF/` directory and updated with custom CRON expressions for any scheduled jobs, extract and customize the file again so as to retain the previous CRON expressions.

Do not replace the `jobs.xml` file on the target release with a previous version because the `jobs.xml` file is likely to change with each new version of Oracle Identity Analytics.

Following is a sample CRON expression for user file import:

```
<bean id="usersImportTrigger"
class="org.springframework.scheduling.quartz.CronTriggerBean">
    <property name="jobDetail">
        <ref bean="usersImportJob"/>
    </property>
    <property name="cronExpression">
        <value>0 0/5 * * * ?</value>
    </property>
</bean>
```

8. Oracle recommends that you extract the `scheduling-context.xml` file, located in *exploded_rbacx.war_directory*`/WEB-INF/`.

   To retain any previous triggering/scheduling customizations, changes should also be made to the `scheduling-context.xml` file for the target release.

   Following is a file sample that would require changes to the `scheduling-context.xml` file in the target release.

```
<property name="jobDetails">
        <list>
                <!--ref bean="usersImportJob"/-->
                <!--ref bean="accountsImportJob"/-->
                <!--ref bean="rolesImportJob"/-->
                <!--ref bean="glossaryImportJob"/-->
                <!--ref bean="policiesImportJob"/-->
                <!--ref bean="businessStructureImportJob"/-->
                <ref bean="identityAuditContinuousViolationScanJob"/>
                <ref bean="identityAuditViolationReminderJob"/>
                <ref bean="certificationReminderJob"/>
                <!--ref bean="reportReminderJob"/-->
                <!--ref bean="stableFolderCleanUpJob"/-->
                <!--ref bean="accountsMaintenanceJob"/-->
                <!--ref bean="roleMembershipRuleJob"/-->
                <ref bean="fullTextIndexMaintenancedJob"/>
                <ref bean="workflowStepSLAJob"/>
                <ref bean="roleStatusAndMembershipMaintenanceJob"/>
                <ref bean="rmPreviewCleanUpJob"/>
                <ref bean="userApplicationMaintenanceJob"/>
                <ref bean="postImportJobsLauncherJob"/>
                <ref bean="certificationRemediationJob"/>
                <ref bean="rmScanArchivalJob"/>
                <ref bean="eventPublishingJob"/>
                <!--<ref bean="rmeRuleMigrationJob"/>-->
                <!--ref bean="identityAuditDataMigrationJob"/-->

        </list>
</property>

<property name="triggers">
        <list>
                <!--ref bean="usersImportTrigger"/-->
                <!--ref bean="accountsImportTrigger"/-->
                <!--ref bean="accountsImportTrigger_2"/-->
                <!--ref bean="accountsImportTrigger_3"/-->
                <!--ref bean="rolesImportTrigger"/-->
```

```
                        <!--ref bean="glossaryImportTrigger"/-->
                        <!--ref bean="policiesImportTrigger"/-->
                        <!--ref bean="businessStructureImportTrigger"/-->
                        <ref bean="identityAuditContinuousViolationScanTrigger"/>
                        <ref bean="identityAuditViolationReminderTrigger"/>
                        <ref bean="certificationReminderTrigger"/>
                        <!--ref bean="reportReminderTrigger"/-->
                        <!--ref bean="stableFolderCleanUpTrigger"/-->
                        <!--ref bean="accountsMaintenanceTrigger"/-->
                        <!--ref bean="roleMembershipRuleTrigger"/-->
                        <ref bean="fullTextIndexMaintenanceTrigger"/>
                        <ref bean="workflowStepSLATrigger"/>
                        <ref bean="roleStatusAndMembershipMaintenanceJobTrigger"/>
                        <ref bean="rmPreviewCleanUpJobTrigger"/>
                        <ref bean="userApplicationMaintenanceTrigger"/>
                        <ref bean="postImportJobsLauncherTrigger"/>
                        <ref bean="certificationRemediationTrigger"/>
                        <ref bean="rmScanArchivalJobTrigger"/>
                        <ref bean="eventPublishingJobTrigger"/>
                        <!--<ref bean="rmeRuleMigrationJobTrigger"/>-->
                        <!--ref bean="identityAuditDataMigrationTrigger"/-->

                </list>
        </property>
```

7

Preparing to Upgrade Oracle Identity Analytics

This chapter contains the following sections:

In this phase of the upgrade process you will document your existing environment, document any custom components, and choose an upgrade version.

7.1 Documenting the Existing Environment

Document your existing environment, including the following items:

1. **Application servers** - Record the application server name and version number, including any service packs.

   Also, note the following:

   a. The operating system version number, including any service packs.

   b. The version number of the Java Development Kit (JDK) installed on the application server.

2. **Database servers** - Record the database server name and version number and any additional service packs.

3. **Supported resources** - Record the names of all supported resources, including version numbers, and any installed service packs. Supported resources also include integrated provisioning servers, such as Oracle Identity Manager.

4. **Web servers** - Record the name and version number of any web servers, including any additional service packs.

7.2 Documenting the Oracle Identity Analytics Installation

Document your Oracle Identity Analytics installation, including the release and build version number.

7.2.1 To Obtain the Oracle Identity Analytics Release and Build Version

The Oracle Identity Analytics release version information can be obtained using any of the following steps.

- To obtain version information from the `rbacx.log` file, do the following:
  1. Start Oracle Identity Analytics.
  2. Open the `rbacx.log` file located in the *$RBACX_HOME*`/logs` folder.
  3. Search the log for the Oracle Identity Analytics version information. The Oracle Identity Analytics version number is written to the log when Oracle Identity Analytics is started, for example:

     ```
     Oracle Identity Analytics (build: 5.0.0.200910016491-GA) Started
     ```
- To obtain version information from `VERSION.txt` within the `.war` file, do the following:
  1. Navigate to the *exploded_rbacx.war_directory*`/WEB-INF/classes` directory.
  2. Locate `VERSION.txt` and view the contents using a text-editor.
  3. Search for the `Implementation-Version` tag, which contains the release version number.

     For example, `Implementation-Version=5.0.0.200910016491-GA`.
- To obtain version information from `MANIFEST.MF` within the `.war` file, do the following:
  1. Navigate to *exploded_rbacx.war_directory*`/META-INF` directory.
  2. Using a text-editor, open the `MANIFEST.MF` file.
  3. Search for the `Implementation-Version` tag, which contains the release version number.

     For example, `Implementation-Version=5.0.0.200910016491-GA`.
- To obtain only the version number, do the following:
  1. Log on to the Oracle Identity Analytics user interface.
  2. Click **About** in the top-right corner of the screen.

     A window containing the version number opens, for example `Version 11.1.1.5.0`.

7.3 Documenting Custom Components

Document any custom components, including the following:

- Custom File-System Objects
- Custom Repository Objects

7.3.1 Documenting Custom File-System Objects

Oracle recommends upgrading custom file-system objects. File-system objects that may have been customized include the following:

- **Modified `iam.properties` file** - Record any changes made to the default `iam.properties` file.

- **Modified `jdbc.properties` file** - Record any changes made to the default `jdbc.properties` file.
- **Modified `ldap.properties` file** - Record any changes made to the default `ldap.properties` file.
- **Modified `rm_idm_init.xml` file** - Record any changes made to the default `rm_idm_init.xml` file.
- **Modified `log4j.properties` file** - Record any changes made to default `log4j.properties` file for customized loggings.
- **Modified `rbacxmessages.properties` file** - Record any changes made to the default `rbacxmessages.properties` file.
- **Modified `rbacxaudit-messages.properties` file** - Record any changes made to the default `rbacxaudit-messages.properties` file.
- **Customized property files** - Record any changes made to other property files on the system.
- **Customized resource adapters (and any other custom Java code)** - Customized resource adapters may require recompilation depending on the target Oracle Identity Analytics version. All custom Java that uses Oracle Identity Analytics APIs (including custom resource adapters) requires a recompilation as part of an upgrade. Also, consider other Java classes that may use the Oracle Identity Analytics library.
- **Custom utilities** - Custom Java utilities, pre-processors, and data parsers may require review and recompilation depending on the target Oracle Identity version. Plan to re-evaluate the code in these scripts to ensure that they align with database schema changes or application modifications.
- **Modified JavaServer Pages (JSP Files)** - Recent Oracle Identity Analytics versions may include API changes. If JSP files were modified during installation, they need to be updated when upgrading. Any JSP file that shipped with the product and was modified during a deployment (or any custom JSP file that uses Oracle Identity Analytics APIs) should be updated so that it is compatible with the JSP structure and API in the target release.

7.3.2 Documenting Custom Repository Objects

Oracle recommends upgrading custom repository objects. Repository objects that may have been customized include the following:

- **Modified E-mail Templates** - Custom e-mail templates may require an export to take advantage of current product enhancements.
- **Custom Repository Schema -** If a schema change occurred since Oracle Identity Analytics was last installed or upgraded, a schema update is required.
- **Custom Reports**
  - Changes made to any reports in the *$RBACX_HOME*`/reports` directory should be incorporated into the reports present in the latest version of Oracle Identity Analytics.
  - Do not overwrite reports in the target release with previous versions because any changes made to the Oracle Identity Analytics schema will cause the reports to break.
  - Custom reports uploaded to Oracle Identity Analytics may require modification in order to comply with schema changes.

- **Custom Workflow Steps** - Document all custom workflow steps created during Oracle Identity Analytics implementation. These steps would need to be recreated in the newer version of Oracle Identity Analytics.
- **Custom CRON Expression** - Custom CRON expressions must be included in the latest version of the Oracle Identity Analytics `jobs.xml` file, otherwise the CRON expressions will be lost.
- **Custom Import Triggers** - To retain previous triggering/scheduling customizations, changes should also be made in the target release `scheduling-context.xml` file

> **Note:** Any occurrence of REPLACE_ME needs to be changed to FILE_SERVER for feeds imported through the Oracle Identity Analytics file importer module.

7.4 Choosing the Upgrade Version

After completing the inventory assessment, choose the version of Oracle Identity Analytics that fulfills your requirements. You may want the latest version, or you may want a more mature version that has service packs available. It is a recommended that you upgrade to the most recent Oracle Identity Analytics release that is available during your testing time frame.

After choosing the target Oracle Identity Analytics release, verify that your current platform is supported.

Also, do the following:

- Read the Release Notes for the target Oracle Identity Analytics release. Pay special attention to the Known Issues section to determine if you need to upgrade the operating system, the JDK software, the application server, or any resources.

8

Upgrading Oracle Identity Analytics in a Test Environment

This chapter contains the following sections:

8.1 Create a Test Environment

Create a test environment similar to the production environment. Replicate the following items from the production environment using the same hardware and software versions:

- Application server
- Database server
- Web server (optional component)
- Common client machine with the corporate image and browser
- Resources and other integrated applications
- Oracle Identity Analytics version and configuration

8.2 Verify That Your Environment is Properly Configured for OIA

In replicating your production environment you should have set the `RBACX_HOME` environment variable and configured your JVM options. Before you upgrade, verify that these important settings are correct in your test environment.

8.2.1 Verify the RBACX_HOME Environment Variable Setting

Before upgrading Oracle Identity Analytics verify that the *RBACX_HOME* environment variable has been permanently created and is set to the directory where you will install Oracle Identity Analytics. To create a permanent environment variable, refer to your operating system documentation for instructions.

Note:

- A permanent *$RBACX_HOME* environment variable should be created under the application server's owner profile. Oracle Identity Analytics deployment will fail if *RBACX_HOME* is declared in a profile inaccessible by the application server.
- For a clustered deployment, the *$RBACX_HOME* environment variable needs to be created on every cluster member.

8.2.2 Verify the JVM Options

Refer to your application server's documentation for information about configuring JVM options.

- You should determine your memory need and set values in your application server's JVM accordingly. The recommended memory settings are as follows:

  ```
  -Xmx2048m -Xms2048m
  ```

  Depending on your specific implementation, you might need to increase these recommended values if you face performance issues with the web interface. Keeping a low minimum value minimizes garbage collection, whereas keeping a higher value decreases response time in the web interface.

- IPv4 is required by Oracle Identity Analytics for network communication. If necessary, add the following JVM option to enforce IPv4 preference over IPv6:

  ```
  -Djava.net.preferIPv4Stack=true
  ```

Also, ensure that your OS is configured to use IPv4. Refer to your operating system documentation for instructions to enable the IPv4 stack.

8.3 Create Upgrade Directories and Set Environment Variables

Complete these steps in preparation for upgrading your software.

8.3.1 Create the Directories Needed During the Upgrade Process

1. Create a directory named `OIA_Upgrade`.

   For example:

   - **Windows -** `C:\OIA_Upgrade`
   - **UNIX -** `/opt/OIA_Upgrade`

2. Unpack the Oracle Identity Analytics installation package (the upgrade software) to the OIA_Upgrade directory.

   - **Windows:**

     `unzip` *oia_install_package.zip* `-d OIA_Upgrade`

   - **UNIX:**

`unzip` *oia_install_package.zip* `-d OIA_Upgrade`

3. Verify that the Oracle Identity Analytics folder structure was properly created.

   The directory should consist of at least the following folders and files.

Table 8–1 Contents of the Base OIA Directory

Name	Type
`.indexes`	*folder*
`conf`	*folder*
`db`	*folder*
legal	*folder*
`rbacx.war`	*file*
`reports`	*folder*
`sample`	*folder*

4. Create a directory named `OIA_Temp`.

   This is the directory to which you will later extract the `rbacx.war` file. Here you will make configuration changes (as needed) prior to recreating the `rbacx.war` file.

   For example:

   - **Windows -** `C:\OIA_Temp`
   - **UNIX -** `/opt/OIA_Temp`

5. Create a directory named `OIA_Lib`.

   Later in Section 8.4, "Download Third-party Library Files" you will copy or download external and third-party files to this location.

   For example:

   - **Windows -** `C:\OIA_Lib`
   - **UNIX -** `/opt/OIA_Lib`

8.3.2 Set the Environment Variable Needed During the Upgrade Process

Set the following environment variables in your test environment.

- **Windows:**

  Type the following commands at a command prompt:

  - `set INSPATH=`*Path to the `OIA_Upgrade` directory*

    For example: `set INSPATH=C:\OIA_Upgrade`

  - `set TEMP=`*Path to the `OIA_Temp` directory*

    For example: `set TEMP=C:\OIA_Temp`

  - `set OIA_LIB=`*Path to the `OIA_Lib` directory*

    For example: `set OIA_LIB=C:\OIA_Lib`

  - `set RBACXWAR=`*Path to the Oracle Identity Analytics deployment directory*

    This is where the updated `rbacx.war` file you create will be placed.

    For example: `set RBACXWAR=`*application server install directory*`\webapps`

- **UNIX:**

  Type the following commands at a command prompt:

  - `export INSPATH=`*Path to the `OIA_Upgrade` directory*

    For example: `export INSPATH=/opt/OIA_Upgrade`

  - `export TEMP=`*Path to the `OIA_Temp` directory*

    For example: `export TEMP=/opt/OIA_Temp`

  - `export OIA_LIB=`*Path to the `OIA_Lib` directory*

    For example: `export OIA_LIB=/opt/OIA_Lib`

  - `export RBACXWAR=`*Path to the Oracle Identity Analytics deployment directory*

    This is where the updated `rbacx.war` file you create will be placed.

    For example: `export RBACXWAR=`*application server install directory*`/webapps`

8.4 Download Third-party Library Files

Prior to upgrading, download or copy the files in this section. Save the files to the `OIA_LIB` library folder that you created in Section 8.3.1, "Create the Directories Needed During the Upgrade Process."

8.4.1 JDBC Drivers

For Oracle Database Server, download from the Oracle website the `ojdbc5.jar` driver if using JDK 1.5, and download the `ojdbc6.jar` driver if using JDK 1.6. The JDBC driver file you choose needs to support both the JDK version you are running, as well as the backend Oracle database instance version.

8.4.2 The jasper-jdt.jar File

This file is required by the Oracle Identity Analytics certification and reporting feature.

Download the `jasper-jdt.jar` file and paste it to the *OIA_LIB* folder:

`http://tomcat.apache.org/dev/dist/m2-repository/org/apache/tomcat/jasper-jdt/6.0.18/`

8.4.3 The CloverETL Library

OIA uses CloverETL for data import and export transformations.

Download the CloverETL Engine class files, version 1.8.1 from this site:

`http://download.berlios.de/cloveretl/cloverETL.rel-1-8-1.zip`

Next, complete the following steps to convert the library file to a JAR file.

8.4.3.1 To Create the CloverETL Library JAR File

Before You Begin - You will need the `unzip` utility and at least a Java 5 JDK.

1. Create a working directory named `files` and then open the directory:

   ```
   mkdir files
   cd files
   ```

2. Check the integrity of the `.zip` file and then expand it:

```
unzip -tq ../cloverETL.rel-1-8-1.zip

unzip -q ../cloverETL.rel-1-8-1.zip
```

3. Remove the `log4j.properties` file to prevent a file conflict in OIA:

   ```
   rm log4j.properties
   ```

4. Go to the parent directory:

   ```
   cd ..
   ```

5. Create the JAR manifest input file `clover.mf`, which consists of these lines:

   ```
   Implementation-Version: 1.8.1
   Implementation-Title: jETeL/Clover
   Implementation-URL:
   http://download.berlios.de/cloveretl/cloverETL.rel-1-8-1.zip
   Implementation-Vendor-Id: org.jetel
   ```

6. Create the JAR file:

   ```
   jar -cf cloverETL-1.8.1.jar -m clover.mf -C files .
   ```

7. Copy the `cloverETL-1.8.1.jar` file to the *OIA_LIB* folder.

8.4.4 The jxl-2.5.9.jar File

OIA uses the Java-Excel API to import data from an Excel spreadsheet file.

Download the `jxl-2.5.9.jar` file from this site:

http://www.andykhan.com/jexcelapi/

8.4.5 The Web Services Description Language for Java Toolkit (WSDL4J)

OIA uses the Web Services Description Language for Java Toolkit (`WSDL4J.jar`) for provisioning server integration, among other things. If you are using Oracle Identity Analytics Web Services, download the WSDL4J JAR file.

8.4.5.1 To Download and Extract the WSDL4J JAR File

1. Download the `wsdl4j-bin-1.6.1.zip` file from this site:

   http://sourceforge.net/projects/wsdl4j/files/WSDL4J/1.6.1/

   Next, follow these steps to extract the `wsdl4j.jar` file and copy it to the `OIA_LIB` folder.

2. Verify the zip file was downloaded without errors:

   - **Windows:**

     ```
     unzip -tq wsdl4j-bin-1.6.1.zip
     ```

   - **UNIX:**

     ```
     unzip -tq wsdl4j-bin-1.6.1.zip
     ```

3. Extract the `wsdl4j.jar` file:

   - **Windows:**

     ```
     unzip -q wsdl4j-bin-1.6.1.zip wsdl4j-1_6_1\lib\wsdl4j.jar
     ```

   - **UNIX:**

     ```
     unzip -q wsdl4j-bin-1.6.1.zip wsdl4j-1_6_1/lib/wsdl4j.jar
     ```

4. Copy the JAR file to the `OIA_LIB` folder and rename it to include the version number:
   - **Windows:**

     ```
     move wsdl4j-1_6_1\lib\wsdl4j.jar %OIA_LIB%\wsdl4j-1.6.1.jar
     ```

   - **UNIX:**

     ```
     mv wsdl4j-1_6_1/lib/wsdl4j.jar $OIA_LIB/wsdl4j-1.6.1.jar
     ```

5. Clean up:
   - **Windows:**

     ```
     del /F wsdl4j-bin-1.6.1.zip  wsdl4j-1_6_1\
     ```

   - **UNIX:**

     ```
     rm -fr  wsdl4j-bin-1.6.1.zip  wsdl4j-1_6_1/
     ```

8.5 Prepare to Upgrade the Test Environment

Before upgrading the Oracle Identity Analytics installation in the test environment, do the following:

Before You Begin - Shut down the Oracle Identity Analytics instance running on the system before backing up your files.

1. **Back up the current Oracle Identity Analytics installation**

   Before upgrading, back up the Oracle Identity Analytics database and the directory where Oracle Identity Analytics is installed. Use third-party backup software or a backup utility supplied with the operating system to back up the Oracle Identity Analytics file system. To back up the database, refer to the documentation provided by your database vendor.

2. **Back up customized Oracle Identity Analytics repository objects**

   Oracle Identity Analytics provides a set of database objects, such as workflow task definitions, that are usually customized for an environment. The upgrade process replaces some of these objects in the database after saving them in the file system.

3. **Verify the environment**

   Before upgrading Oracle Identity Analytics in the test environment, verify the following:

   - All servers are present on the network
   - The schema is up-to-date
   - The location of the Oracle Identity Analytics application is correct

8.6 Upgrade the OIA Schema on the Database Server

Before you upgrade the Oracle Identity Analytics software, upgrade the schema on the database server.

This section is organized as follows:

- Section 8.6.1, "To Upgrade the OIA Schema on the Database Server on a Windows Platform"
- Section 8.6.2, "To Upgrade the OIA Schema on the Database Server on a UNIX Platform"

8.6.1 To Upgrade the OIA Schema on the Database Server on a Windows Platform

Before You Begin - You should have created a backup of the Oracle Identity Analytics database.

Oracle Database

You can upgrade the Oracle Identity Analytics schema using either the command prompt or the Oracle iSQL Plus Web Console (*available in Oracle 10g Database Server*). If the Oracle Database Server is not installed locally, use the iSQL Plus Web Console to upgrade the schema.

- To upgrade the schema from a command prompt, follow these steps:
  1. Stop the application server running Oracle Identity Analytics.
  2. Navigate to the directory containing the upgrade scripts (`C:\OIA_Upgrade\db\oracle`) by typing:

     ```
     C:\> cd C:\%OIA_Upgrade%\db\oracle
     ```

  3. Run the following command(s) to execute the upgrade script(s):

     ```
     C:\> sqlplus rbacxservice/rbacxservice
     @rbacx-previous_version_To_current_version_oracle.sql
     ```

- To create the schema using the iSQL Plus Web Console, follow these steps:
  1. Open the following URL in a web browser. The default port for the iSQL Plus Web Console is 5560.

     ```
     http://hostname:5560/isqlplus
     ```

  2. Select **Load Script** and browse to `C:\%OIA_Upgrade%\db\oracle`.
  3. Locate the file `rbacx-previous_version_To_current_version_oracle.sql` and click **Load**.

     Note: For Oracle Identity Analytics 11gR1 PS1 the version number is 11.1.1.5.0.

     The Oracle Identity Analytics upgrade schema creation script is loaded into the workspace window.
  4. Click **Execute**.

     The script is executed and the Oracle Identity Analytics schema is upgraded on the system.

8.6.2 To Upgrade the OIA Schema on the Database Server on a UNIX Platform

Before You Begin - You should have created a backup of the Oracle Identity Analytics database.

Oracle Database

You can upgrade the Oracle Identity Analytics schema using either the command prompt or the Oracle iSQL Plus Web Console (*available in Oracle 10g Database Server*). If the Oracle Database Server is not installed locally under the `/opt` directory, use the iSQL Plus Web Console to upgrade the schema.

Note: For Oracle Identity Analytics 11gR1 PS1 the version number is 11.1.1.5.0.

- To upgrade the schema from a command prompt, follow these steps:
  1. Stop the application server running Oracle Identity Analytics.
  2. Navigate to the directory containing the upgrade scripts (`/opt/`*OIA_Upgrade*`/db`) by typing:

     ```
     $ su - oracle
     ```

     ```
     $ export ORACLE_HOME=/opt/oracle/Path-to-Oracle-Database-Install-Directory
     ```

     ```
     $ cd /opt/OIA_Upgrade/db/oracle
     ```
  3. Run the following command(s) to execute the upgrade script(s):

     ```
     $ ./sqlplus rbacxservice/rbacxservice
     @rbacx-previous_version_To_current_version_oracle.sql
     ```

     The Oracle Identity Analytics database schema is upgraded on Oracle Database server.
- To create the schema using the iSQL Plus Web Console, follow these steps:
  1. Open the following URL in a web browser. The default port for the iSQL Plus Web Console is 5560.

     ```
     http://hostname:5560/isqlplus
     ```
  2. Select **Load Script** and browse to `/opt/`*OIA_Upgrade*`/db/oracle`.
  3. Locate the file `rbacx-`*previous_version*`_To_`*current_version*`_oracle.sql` and click **Load**.

     The Oracle Identity Analytics schema upgrade script is loaded into the workspace window.
  4. Click **Execute**.

     The script is executed and the Oracle Identity Analytics schema is upgraded on the system.

8.7 Upgrade the Oracle Identity Analytics Environment

After upgrading the schema on the database server, upgrade your Oracle Identity Analytics software.

This section is organized as follows:

- Section 8.7.1, "To Upgrade the Environment on a Windows Platform for a Standalone Deployment"
- Section 8.7.2, "To Upgrade the Environment on a UNIX Platform for a Standalone Deployment"
- Section 8.7.3, "To Upgrade the Environment on a Windows Platform for a Clustered Deployment"
- Section 8.7.4, "To Upgrade the Environment on a UNIX Platform for a Clustered Deployment"

8.7.1 To Upgrade the Environment on a Windows Platform for a Standalone Deployment

Use the following steps to upgrade Oracle Identity Analytics on a supported Windows platform.

1. Stop the application server.
2. Clear the cache on your application server.

   For instructions, consult your application server documentation.

   Note: On WebLogic, delete the `cache` and `tmp` folders.

   If there is a `stage` folder, delete `\stage\rbacx` as well. Or, if you changed the default application name, delete `\stage\`*nameOfYourOiaApp*.

3. Upgrade the report-related files in the *RBACX_HOME*`\reports` directory:
   a. Remove the existing files from the *RBACX_HOME*`\reports` directory:

      ```
      rmdir /s /q %RBACX_HOME%\reports
      ```

   b. Copy the reports that are appropriate for your database to the *RBACX_HOME*`\reports` folder.

      The following sample steps assume that the target database server is Oracle Database.

      ```
      xcopy %INSPATH%\reports %RBACX_HOME%\reports /I/E
      ```

      ```
      xcopy %RBACX_HOME%\reports\oracle * %RBACX_HOME%\reports
      ```

4. Upgrade the search-related files in the *RBACX_HOME*`\.indexes` directory:
   a. Remove search-related files from the *RBACX_HOME*`\.indexes` directory:

      ```
      rmdir /s /q %RBACX_HOME%\.indexes
      ```

   b. Copy the files related to advanced search from the upgrade folder to the `RBACX_HOME\.indexes folder`:

      ```
      xcopy %INSPATH%\.indexes %RBACX_HOME%\.indexes /I/E
      ```

5. Make changes to the following files located under the `%INSPATH%\conf` directory to reflect customizations as documented in Section 7.3, "Documenting Custom Components."
   - **iam.properties** - In a text editor, replace any occurrences of *$RBACX_HOME* with the Oracle Identity Analytics installation directory path.
   - **jdbc.properties** -
     - In a text editor, edit the following lines, substituting *$SERVER_NAME* and *$PORT_NUMBER* with the host name and connectivity port of the target database.
     - Copy the JDBC account password from your current `jdbc.properties` file (`%RBACX_HOME%\conf\jdbc.properties`) to the new `%INSPATH%\conf\jdbc.properties` file. The JDBC account password for the default user `rbacxservice` is stored in the Oracle Identity Analytics `jdbc.properties` file as property `jdbc.password.encrypted`.

       For example:

```
jdbc.password.encrypted=d579601be2566d0095865af775bf7371
```

Copying the JDBC account password is required to ensure that OIA starts successfully.

- **ldap.properties** - Refer to the "Authenticating With LDAP" chapter in the *System Integrator's Guide for Oracle Identity Analytics* for information about LDAP customizations.
- **rm_idm_init.xml** - This file (the SPML exchange file) is required by Oracle Identity Analytics to exchange and manage information with Oracle Waveset (Sun Identity Manager).

6. Replace the files in the RBACX_HOME\conf directory with the files you changed in the previous step:
   a. Remove the conf directory from *RBACX_HOME*:

      ```
      rmdir /s /q %RBACX_HOME%\conf
      ```

   b. Copy the upgraded conf directory files to the RBACX_HOME directory:

      ```
      xcopy %INSPATH%\conf %RBACX_HOME%\conf /I/E
      ```

7. Expand the rbacx.war file.

   ```
   cd %TEMP%
   ```

   ```
   jar -xvf %INSPATH%\rbacx.war
   ```

   In the following steps you will make configuration changes to the expanded rbacx.war file.

8. If you are using CloverETL, enable it in the configuration as follows:
   a. In a text editor, open WEB-INF/etl-context.xml and uncomment the etlManager bean definition.
   b. In a text editor, open WEB-INF/iam-context.xml and uncomment the etlManager bean reference in the property list of the file bean definition.
9. Copy the downloaded third-party library files to the library directory in the expanded WAR file:

   ```
   copy %OIA_LIB% %TEMP%\WEB-INF\lib
   ```

10. Apply any customizations necessary for the environment to the expanded .war file in the %TEMP% directory.
11. Update the log4j.properties file with the correct path for your environment.

    Note: If the Oracle Identity Analytics log file is going to be created in any folder other than the default log folder as defined by the application server, complete the step. Otherwise, skip and go to the next step.

    a. In a text editor, open the log4j.properties file located in the %TEMP%\WEB-INF folder.
    b. Locate the following line under # File Appender.

       ```
       log4j.appender.file.file=logs\rbacx.log
       ```

    c. Replace logs\rbacx.log with the full path to where the log file should be written.

For example, the line should look like this:

```
log4j.appender.file.file=C:\Oracle\OIA_Install\logs\rbacx.log
```

12. Update the `jasper.properties` file.
    a. In a text editor, open the `jasper.properties` file located in the `%TEMP%\WEB-INF\classes` folder.
    b. Add the following line to the end of the file:

       `net.sf.jasperreports.compiler.classpath=`*Path to your rbacx folder*`\rbacx\WEB-INF\lib\jasperreports-2.0.5-javaflow.jar`

       Note: The path to the `rbacx` deployment folder will vary on the application server.

13. Make the following changes if there are multiple instances of Oracle Identity Analytics, standalone or clustered, on the same subnet.
    a. Navigate to the `%TEMP%\WEB-INF` directory.
    b. In a text editor, open `application-context.xml`, find bean ID `commManager`, and examine the `constructor-arg` value.
    c. Set the `constructor-arg` value with a unique instance name, for example `value="OIA-Instance-1"`.
    d. In bean ID `commManager`, locate the `constructor-arg index="1"` value.
    e. Replace the value with the IP address of each cluster member. This setting binds the multicast addresses to the IP addresses. In addition, add the `enabled` property and set it to `true`. For example:

       ```
       <constructor-arg index="1" value="140.84.134.133;140.84.135.88"/>
       <property name="enabled" value="true"/>
       ```

       Save the `application-context.xml` file.
    f. In a text editor, open `search-context.xml`, find bean ID `searchConfiguration`, and examine the `constructor-arg` value.

       If this is a standalone deployment, set the `constructor-arg` defaults to a value of 0, which is specified as `value="0"`.
    g. Navigate to the `%TEMP%\WEB-INF\classes` directory and do the following:
       – In a text editor, open `oscache.properties` (located in the `%TEMP%\WEB-INF\classes` directory), and find the `cache.cluster.multicast.ip` property.
       – Uncomment `cache.cluster.multicast.ip` by removing the # at the start of the line. Each Oracle Identity Analytics instance requires a unique `cache.cluster.multicast.ip` value.
       – Uncomment the following line by removing the # at the start of the line.

         ```
         cache.event.listeners=com.opensymphony.oscache.plugins.clustersupport.J
         avaGroupsBroadcastingListener,com.opensymphony.oscache.extra.CacheMapAc
         cessEventListenerImpl
         ```

14. If you are using OIA Web Services, uncomment its configuration.

    See the *API Guide for Oracle Identity Analytics*, "Enabling Web Services."

15. Repackage the `.war` file in the `%TEMP%` directory if changes are made. Type the following commands in the test environment:

    ```
    cd %TEMP%

    jar -cvfM %RBACXWAR%\rbacx.war .
    ```

16. Uninstall Oracle Identity Analytics from the application server.
17. Deploy the rebuilt Oracle Identity Analytics `.war` file on the target application server.

    Refer to Chapter 4, "Deploying Oracle Identity Analytics" for instructions about deploying Oracle Identity Analytics.
18. Start the application server.
19. Go to the following address in a browser:

    `https://`*Hostname*`:`*Port-Number*`/rbacx/welcome.action`

    When the Welcome screen appears, enter your rbacxadmin user name and password and verify that the installation is successful.
20. Open the `rbacx.log` file and check for errors.

    The installation is successful if a message similar to the following appears in the `rbacx.log` file:

    ```
    Oracle Identity Analytics(build: 11.1.1.5.0.201107nn_nn_nnnn) Started
    ```

8.7.2 To Upgrade the Environment on a UNIX Platform for a Standalone Deployment

Use the following steps to upgrade Oracle Identity Analytics manually on a supported UNIX platform.

1. Stop the application server.
2. Clear the cache on your application server.

   For instructions, consult your application server documentation.

   Note: On WebLogic, delete the `cache` and `tmp` folders.

   If there is a `stage` / `staging` folder, delete `/stage/rbacx` as well. Or, if you changed the default application name, delete `/stage/`*nameOfYourOiaApp*.

   ---

3. Upgrade the report-related files in the *RBACX_HOME*`/reports` directory:
   a. Remove the existing files from the *RBACX_HOME*`/reports` directory:

      ```
      /usr/bin/rm -f $RBACX_HOME/reports
      ```

   b. Copy the reports that are appropriate for your database to the *RBACX_HOME*`/reports` folder.

      The following sample steps assume that the target database server is Oracle Database.

      ```
      cp -R $INSPATH/reports $RBACX_HOME/.

      cp -R $RBACX_HOME/reports/oracle/* $RBACX_HOME/reports/.
      ```

4. Upgrade the search-related files in the *RBACX_HOME*`/.indexes` directory:

a. Remove search-related files from the *RBACX_HOME*/.indexes directory:

```
/usr/bin/rm -f $RBACX_HOME/.indexes
```

b. Copy the files related to advanced search from the upgrade folder to the RBACX_HOME/.indexes folder:

```
cp -R $INSPATH/.indexes $RBACX_HOME/.
```

5. Make changes to the following files located under the $INSPATH/conf directory to reflect customizations as documented in Section 7.3, "Documenting Custom Components."

- **iam.properties** - In a text editor, replace any occurrences of $RBACX_HOME with the Oracle Identity Analytics installation directory path.
- **jdbc.properties** -
  - In a text editor, edit the following lines, substituting *$SERVER_NAME* and *$PORT_NUMBER* with the host name and connectivity port of the target database.
  - Copy the JDBC account password from your current jdbc.properties file ($RBACX_HOME/conf/jdbc.properties) to the new $INSPATH/conf/jdbc.properties file. The JDBC account password for the default user rbacxservice is stored in the Oracle Identity Analytics conf/jdbc.properties file as property jdbc.password.encrypted.

    For example:

    ```
    jdbc.password.encrypted=d579601be2566d0095865af775bf7371
    ```

    OIA will not start if you do not use your existing JDBC account password.
- **ldap.properties** - Refer to the "Authenticating With LDAP" chapter in the *System Integrator's Guide for Oracle Identity Analytics* for instructions on LDAP customizations.
- **rm_idm_init.xml** - This file is required by Oracle Identity Analytics to exchange and manage information with Identity Manager.

6. Replace the files in the RBACX_HOME/conf directory with the files you changed in the previous step:

a. Remove the conf directory from *RBACX_HOME*:

```
/usr/bin/rm -f $RBACX_HOME/conf
```

b. Copy the upgraded conf directory files to the RBACX_HOME directory:

```
cp -R $INSPATH/conf $RBACX_HOME/.
```

7. Expand the rbacx.war file.

```
cd $TEMP
```

```
jar xvf $INSPATH/rbacx.war
```

In the following steps you will make configuration changes to the expanded rbacx.war file.

8. If you are using CloverETL, enable it in the configuration as follows:

a. In a text editor, open WEB-INF/etl-context.xml and uncomment the etlManager bean definition.

b. In a text editor, open `WEB-INF/iam-context.xml` and uncomment the `etlManager` bean reference in the property list of the file bean definition.

9. Copy the downloaded third-party library files to the library directory in the expanded WAR file:

   ```
   cp $OIA_LIB/* $TEMP/WEB-INF/lib
   ```

10. Apply any customizations necessary for the environment to the extracted `.war` file in the `$TEMP` directory.

11. Update the `log4j.properties` file with the correct path for your environment.

    Note: If the Oracle Identity Analytics log file is going to be created in any folder other than the default log folder as defined by the application server, complete the step. Otherwise, skip and go to the next step.

    a. In a text editor, open the `log4j.properties` file located in the `$TEMP/WEB-INF` folder.

    b. Locate the following line under `# File Appender`.

       ```
       log4j.appender.file.file=logs/rbacx.log
       ```

    c. Replace `logs/rbacx.log` with the full path to where the log file should be written.

       For example, the line should look like this:

       ```
       log4j.appender.file.file=/opt/Oracle/OIA_Install/logs/rbacx.log
       ```

12. Update the `jasper.properties` file.

    a. In a text editor, open the `jasper.properties` file located in the `WEB-INF/classes` folder.

    b. Add the following line to the end of the file:

       `net.sf.jasperreports.compiler.classpath=`*Path to your rbacx folder*`/rbacx/WEB-INF/lib/jasperreports-2.0.5-javaflow.jar`

       Note: The path to the `rbacx` deployment folder will vary on the application server.

13. Make the following changes if there are multiple instances of Oracle Identity Analytics, standalone or clustered, on the same subnet.

    a. Navigate to the `$TEMP/WEB-INF` directory.

    b. In a text editor, open `application-context.xml`, find bean ID `commManager`, and examine the `constructor-arg` value.

    c. Set the `constructor-arg` value with a unique instance name, for example `value="OIA-Instance-1"`.

    d. In a text editor, open `search-context.xml`, find bean ID `searchConfiguration`, and examine the `constructor-arg` value.

       If this is a standalone deployment, set the `constructor-arg` defaults to a value of 0, which is specified as `value="0"`.

e. In bean ID `commManager`, locate the `constructor-arg index="1"` value.

f. Replace the value with the IP address of each cluster member. This setting binds the multicast addresses to the IP addresses. In addition, add the `enabled` property and set it to `true`. For example:

```
<constructor-arg index="1" value="140.84.134.133;140.84.135.88"/>
<property name="enabled" value="true"/>
```

Save the application-context.xml file.

g. Navigate to the `$TEMP/WEB-INF/classes` directory and do the following:

- In a text editor, open `oscache.properties` (located in the `$TEMP/WEB-INF/classes` directory), and find the `cache.cluster.multicast.ip` property.
- Uncomment `cache.cluster.multicast.ip` by removing the # at the start of the line. Each Oracle Identity Analytics instance requires a unique `cache.cluster.multicast.ip` value.
- Uncomment the following line by removing the # at the start of the line.

```
cache.event.listeners=com.opensymphony.oscache.plugins.clustersupport.J
avaGroupsBroadcastingListener,com.opensymphony.oscache.extra.CacheMapAc
cessEventListenerImpl
```

14. If you are using OIA Web Services, uncomment its configuration.

    See the *API Guide for Oracle Identity Analytics*, "Enabling Web Services."

15. Repackage the `.war` file in the `$TEMP` directory if you made changes.

    Type the following commands in the test environment:

```
cd $TEMP

jar cvfM $RBACXWAR/rbacx.war .
```

16. Uninstall Oracle Identity Analytics from the application server.

17. Deploy the rebuilt Oracle Identity Analytics `.war` file on the target application server.

    Refer to Chapter 4, "Deploying Oracle Identity Analytics" for more information.

18. Start the application server.

19. Go to the following address in a browser:

    `https://`*Hostname*`:`*Port-Number*`/rbacx/welcome.action`

    When the Welcome screen appears, enter your rbacxadmin user name and password and verify that the installation is successful.

20. Open the `rbacx.log` file and check for errors.

    The installation is successful if a message similar to the following appears in the `rbacx.log` file:

```
Oracle Identity Analytics(build: 11.1.1.5.0.201107nn_nn_nnnn) Started
```

8.7.3 To Upgrade the Environment on a Windows Platform for a Clustered Deployment

Use the following steps to upgrade Oracle Identity Analytics on a supported Windows platform.

1. Stop the application server.
2. Clear the cache on your application servers.

   For instructions, consult your application server documentation.

   Note: On WebLogic, delete the `cache` and `tmp` folders.

   If there is a `stage` (or `staging`) folder, delete `\stage\rbacx` as well. Or, if you changed the default application name, delete `\stage\`*nameOfYourOiaApp*.

   ---

3. Upgrade the report-related files in the *RBACX_HOME*`\reports` directory:
   a. Remove the existing files from the *RBACX_HOME*`\reports` directory:

      ```
      rmdir /s /q %RBACX_HOME%\reports
      ```

   b. Copy the reports that are appropriate for your database to the *RBACX_HOME*`\reports` folder.

      The following sample steps assume that the target database server is Oracle Database.

      ```
      xcopy %INSPATH%\reports %RBACX_HOME%\reports /I/E
      ```

      ```
      xcopy %RBACX_HOME%\reports\oracle * %RBACX_HOME%\reports
      ```

4. Upgrade the search-related files in the *RBACX_HOME*`\.indexes` directory:
   a. Remove search-related files from the *RBACX_HOME*`\.indexes` directory:

      ```
      rmdir /s /q %RBACX_HOME%\.indexes
      ```

   b. Copy the files related to advanced search from the upgrade folder to the RBACX_HOME`\.indexes folder`:

      ```
      xcopy %INSPATH%\.indexes %RBACX_HOME%\.indexes /I/E
      ```

5. Make changes to the following files located under `%INSPATH%\conf` directory to reflect customizations as documented in Section 7.3.1, "Documenting Custom File-System Objects":
   - **iam.properties** - In a text editor, replace any occurrences of `$RBACX_HOME` with the Oracle Identity Analytics installation directory path.

     Note: When the application server is clustered, the nodes can maintain localized import/export directories, or utilize import/export directories on a NFS share. If the nodes use a shared NFS location for import/export, substitute *$RBACX_HOME* with the path to the NFS share.

   - **jdbc.properties**
     - In a text editor, edit the following lines, substituting *$SERVER_NAME* and *$PORT_NUMBER* with the host name and connectivity port of the target database. Refer to Section 3.5, "To Configure Oracle Identity Analytics for Standalone Deployment" for instructions.
     - Make the following change to `jdbc.properties` for clustered Quartz support:

```
jdbc.quartz.isClustered=true
```

- Copy the JDBC account password from your current `jdbc.properties` file (`%RBACX_HOME%\conf\jdbc.properties`) to the new `%INSPATH%\conf\jdbc.properties` file. The JDBC account password for the default user `rbacxservice` is stored in the Oracle Identity Analytics `conf\jdbc.properties` file as property `jdbc.password.encrypted`.

  For example:

  ```
  jdbc.password.encrypted=d579601be2566d0095865af775bf7371
  ```

  Copying the JDBC account password is required to ensure that OIA starts successfully.

- Save the `jdbc.properties` file.

- **ldap.properties** - Refer to the "Authenticating With LDAP" chapter in the *System Integrator's Guide for Oracle Identity Analytics* for instructions on LDAP customizations.
- **rm_idm_init.xml** - This file is required by Oracle Identity Analytics to exchange and manage information with Identity Manager.

6. Replace the files in the RBACX_HOME`\conf` directory with the files you changed in the previous step:
   a. Remove the conf directory from *RBACX_HOME*:

      ```
      rmdir /s /q %RBACX_HOME%\conf
      ```

   b. Copy the upgraded `conf` directory files to the RBACX_HOME directory:

      ```
      xcopy %INSPATH%\conf %RBACX_HOME%\conf /I/E
      ```

7. Expand the `rbacx.war` file.

   ```
   cd %TEMP%
   ```

   ```
   jar -xvf %INSPATH%\rbacx.war
   ```

   In the following steps you will make configuration changes to the expanded `rbacx.war` file.

8. If you are using CloverETL, enable it in the configuration as follows:
   a. In a text editor, open `WEB-INF/etl-context.xml` and uncomment the `etlManager` bean definition.
   b. In a text editor, open `WEB-INF/iam-context.xml` and uncomment the `etlManager` bean reference in the property list of the file bean definition.
9. Copy the downloaded third-party library files to the library directory in the expanded WAR file:

   ```
   copy %OIA_LIB% %TEMP%\WEB-INF\lib
   ```

10. Apply any customizations necessary for the environment to the extracted `.war` file in the `%TEMP%` directory.
11. Update the `log4j.properties` file with the correct path for your environment.

Note: If the Oracle Identity Analytics log file is going to be created in any folder other than the default log folder as defined by the application server, complete the step. Otherwise, skip and go to the next step.

a. In a text editor, open the `log4j.properties` file located in the `%TEMP%\WEB-INF` folder.

b. Locate the following line under `# File Appender`.

```
log4j.appender.file.file=logs\rbacx.log
```

c. Replace `logs\rbacx.log` with the full path to where the log file should be written.

For example, the line should look like this:

```
log4j.appender.file.file=C:\Oracle\OIA_Install\logs\rbacx.log
```

12. Update the `jasper.properties` file.

a. In a text editor, open the `jasper.properties` file located in the `%TEMP%\WEB-INF\classes` folder.

b. Add the following line to the end of the file:

`net.sf.jasperreports.compiler.classpath=`*Path to your rbacx folder*`\rbacx\WEB-INF\lib\jasperreports-2.0.5-javaflow.jar`

Note: The path to the `rbacx` deployment folder will vary on the application server.

13. Make the following changes to enable Oracle Identity Analytics support for clustered application server deployments:

a. Navigate to the `%TEMP%\WEB-INF` directory.

b. In a text editor, open `application-context.xml`, find bean ID `commManager`, and examine the `constructor-arg` value.

c. Set the `constructor-arg` value as the cluster name---for example, `value="Prod-1-Cluster"`.

- If Oracle Identity Analytics is deployed on multiple clusters within the same subnet, you should define unique `constructor-arg` values for each deployment. For example, if both clusters Prod-Cluster and QA-Cluster have Oracle Identity Analytics deployed, the `constructor-arg` values of each should be set to `Prod-Cluster` and `QA-Cluster` respectively.
- Members of the same cluster should have the same `constructor-arg` value.

d. In bean ID `commManager`, locate the `constructor-arg index="1"` value.

e. Replace the value with the IP address of each cluster member. This setting binds the multicast addresses to the IP addresses. In addition, add the `enabled` property and set it to `true`. For example:

```
<constructor-arg index="1" value="140.84.134.133;140.84.135.88"/>
<property name="enabled" value="true"/>
```

Save the application-context.xml file.

f. In a text editor, open `search-context.xml`, find bean ID `searchConfiguration`, and examine the `constructor-arg` value.

For a clustered deployment, `constructor-arg` defaults to a value of 1 or 2 depending on the location of the `.indexes` directory.

To set the constructor-arg value, do the following:

- If each clustered node will be accessing *local* individual `.indexes` directories, set `constructor-arg` to 1. For example, `value="1"`.
- If clustered nodes will be accessing *a shared* `.indexes` directory, set `constructor-arg` to 2. For example, `value="2"`. The `.indexes` directory needs to be located on an NFS share location where each clustered node has read-write permission. Edit `indexLocation` such that the NFS share location replaces *$RBACX_HOME* in the value field.

g. If multiple instances of Oracle Identity Analytics, standalone or clustered, exist on the same subnet, navigate to the `%TEMP%\WEB-INF\classes` directory and do the following:

- In a text editor, open `oscache.properties` (located in the `%TEMP%\WEB-INF\classes` directory), and find the `cache.cluster.multicast.ip` property.
- Uncomment `cache.cluster.multicast.ip` by removing # at the start of the line. Each non-member instance requires a unique `cache.cluster.multicast.ip` value.
- Uncomment the following line by removing the # at the start of the line.

```
cache.event.listeners=com.opensymphony.oscache.plugins.clustersupport.J
avaGroupsBroadcastingListener,com.opensymphony.oscache.extra.CacheMapAc
cessEventListenerImpl
```

14. If you are using OIA Web Services, uncomment its configuration.

See the *API Guide for Oracle Identity Analytics*, "Enabling Web Services."

15. Repackage the `.war` file in the `%TEMP%` directory if changes are made. Type the following commands in the test environment:

```
cd %TEMP%
```

```
jar -cvfM %RBACXWAR%\rbacx.war .
```

16. Uninstall Oracle Identity Analytics from the application server.

17. Deploy the rebuilt Oracle Identity Analytics `.war` file on the target application server.

Refer to Chapter 4, "Deploying Oracle Identity Analytics" for instructions about Oracle Identity Analytics deployment.

18. Start the application server.

19. Go to the following address in a browser:

`https://`*Hostname*`:`*Port-Number*`/rbacx/welcome.action`

When the Welcome screen appears, enter your rbacxadmin user name and password and verify that the installation is successful.

20. Open the `rbacx.log` file and check for errors.

The installation is successful if a message similar to the following appears in the `rbacx.log` file:

```
Oracle Identity Analytics build: 11.1.1.5.0.201107nn_nn_nnnn) Started
```

8.7.4 To Upgrade the Environment on a UNIX Platform for a Clustered Deployment

Use the following steps to upgrade Oracle Identity Analytics manually on a supported UNIX platform.

1. Stop the application servers.
2. Clear the cache on your application servers.

   For instructions, consult your application server documentation.

   Note: On WebLogic, delete the `cache` and `tmp` folders.

   If there is a `stage / staging` folder, delete `/stage/rbacx` as well. Or, if you changed the default application name, delete `/stage/`*nameOfYourOiaApp*.

   ---

3. Upgrade the report-related files in the *RBACX_HOME*`/reports` directory:
   a. Remove the existing files from the *RBACX_HOME*`/reports` directory:

      ```
      /usr/bin/rm -f $RBACX_HOME/reports
      ```

   b. Copy the reports that are appropriate for your database to the *RBACX_HOME*`/reports` folder.

      The following sample steps assume that the target database server is Oracle Database.

      ```
      cp -R $INSPATH/reports $RBACX_HOME/.
      ```

      ```
      cp -R $RBACX_HOME/reports/oracle/* $RBACX_HOME/reports/.
      ```

4. Upgrade the search-related files in the *RBACX_HOME*`/.indexes` directory:
   a. Remove search-related files from the *RBACX_HOME*`/.indexes` directory:

      ```
      /usr/bin/rm -f $RBACX_HOME/.indexes
      ```

   b. Copy the files related to advanced search from the upgrade folder to the RBACX_HOME`/.indexes folder`:

      ```
      cp -R $INSPATH/.indexes $RBACX_HOME/.
      ```

5. Make changes to the following files located under the `$INSPATH/conf` directory to reflect customizations as documented in Section 7.3, "Documenting Custom Components."

   - **iam.properties** - In a text editor, replace any occurrences of `$RBACX_HOME` with the Oracle Identity Analytics installation directory path.

     Note: When the application server is clustered, the nodes can maintain localized import/export directories, or utilize import/export directories on a NFS share. If the nodes use a shared NFS location for import/export, substitute *$RBACX_HOME* with the path to the NFS share.

- **jdbc.properties** -
  - In a text editor, edit the following lines, substituting `$SERVER_NAME` and `$PORT_NUMBER` with the host name and connectivity port of the target database. Refer to Section 3.5, "To Configure Oracle Identity Analytics for Standalone Deployment" for instructions.
  - Copy the JDBC account password from your current `jdbc.properties` file (`$RBACX_HOME/conf/jdbc.properties`) to the new `$INSPATH/conf/jdbc.properties` file. The JDBC account password for the default user `rbacxservice` is stored in the Oracle Identity Analytics `conf/jdbc.properties` file as property `jdbc.password.encrypted`.

    For example:

    ```
    jdbc.password.encrypted=d579601be2566d0095865af775bf7371
    ```

    Copying the JDBC account password is required to ensure that OIA starts successfully.

  Make the following change to `jdbc.properties` for clustered Quartz support, and save the file:

  ```
  jdbc.quartz.isClustered=true
  ```

- **ldap.properties** - Refer to the "Authenticating With LDAP" chapter in the *System Integrator's Guide for Oracle Identity Analytics* for instructions on LDAP customizations.
- **rm_idm_init.xml** - This file is required by Oracle Identity Analytics to exchange and manage information with Identity Manager.

6. Replace the files in the RBACX_HOME/`conf` directory with the files you changed in the previous step:
   a. Remove the conf directory from *RBACX_HOME*:

      ```
      /usr/bin/rm -f $RBACX_HOME/conf
      ```

   b. Copy the upgraded `conf` directory files to the RBACX_HOME directory:

      ```
      cp -R $INSPATH/conf $RBACX_HOME/.
      ```

7. Expand the `rbacx.war` file.

   ```
   cd $TEMP
   ```

   ```
   jar xvf $INSPATH/rbacx.war
   ```

   In the following steps you will make configuration changes to the expanded `rbacx.war` file.

8. If you are using CloverETL, enable it in the configuration as follows:
   a. In a text editor, open `WEB-INF/etl-context.xml` and uncomment the `etlManager` bean definition.
   b. In a text editor, open `WEB-INF/iam-context.xml` and uncomment the `etlManager` bean reference in the property list of the file bean definition.
9. Copy the downloaded third-party library files to the library directory in the expanded WAR file:

   ```
   cp $OIA_LIB/* $TEMP/WEB-INF/lib
   ```

10. Apply any customizations necessary for the environment to the extracted `.war` file in the `$TEMP` directory.

11. Update the `log4j.properties` file with the correct path for your environment.

    Note: If the Oracle Identity Analytics log file is going to be created in any folder other than the default log folder as defined by the application server, complete the step. Otherwise, skip and go to the next step.

    a. In a text editor, open the `log4j.properties` file located in the `$TEMP/WEB-INF` folder.

    b. Locate the following line under `# File Appender`.

    ```
    log4j.appender.file.file=logs/rbacx.log
    ```

    c. Replace `logs/rbacx.log` with the full path to where the log file should be written.

    For example, the line should look like this:

    ```
    log4j.appender.file.file=/opt/Oracle/OIA_Install/logs/rbacx.log
    ```

12. Update the `jasper.properties` file.

    a. In a text editor, open the `jasper.properties` file located in the `WEB-INF/classes` folder.

    b. Add the following line to the end of the file:

    `net.sf.jasperreports.compiler.classpath=`*Path to your rbacx folder*`/rbacx/WEB-INF/lib/jasperreports-2.0.5-javaflow.jar`

    Note: The path to the `rbacx` deployment folder will vary on the application server.

13. Make the following changes to enable Oracle Identity Analytics support for clustered application server deployments.

    a. Navigate to the `$TEMP/WEB-INF` directory.

    b. In a text editor, open `application-context.xml`, find bean ID `commManager`, and examine the `constructor-arg` value.

    c. Set the `constructor-arg` value as the cluster name---for example, `value="Prod-1-Cluster"`.

    – If Oracle Identity Analytics is deployed on multiple clusters within the same subnet, you should define unique `constructor-arg` values for each deployment. For example, if both clusters Prod-Cluster and QA-Cluster have Oracle Identity Analytics deployed, the constructor-arg values of each should be set to `Prod-Cluster` and `QA-Cluster` respectively.

    – Members of the same cluster should have the same `constructor-arg` value.

    d. In bean ID `commManager`, locate the `constructor-arg index="1"` value.

    e. Replace the value with the IP address of each cluster member. This setting binds the multicast addresses to the IP addresses. In addition, add the `enabled` property and set it to `true`. For example:

    ```
    <constructor-arg index="1" value="140.84.134.133;140.84.135.88"/>
    ```

```
<property name="enabled" value="true"/>
```

Save the application-context.xml file.

f. In a text editor, open `search-context.xml`, find bean ID `searchConfiguration`, and examine the `constructor-arg` value.

For a clustered deployment, `constructor-arg` defaults to a value of 1 or 2 depending on the location of the `.indexes` directory.

To set the `constructor-arg` value, do the following:

- If each clustered node will be accessing *local* individual `.indexes` directories, set `constructor-arg` to 1. For example, `value="1"`.
- If clustered nodes will be accessing *a shared* `.indexes` directory, set `constructor-arg` to 2. For example, `value="2"`. The `.indexes` directory needs to be located on an NFS share location where each clustered node has read-write permission. Edit `indexLocation` such that the NFS share location replaces *$RBACX_HOME* in the value field.

g. If multiple instances of Oracle Identity Analytics, standalone or clustered, exist on the same subnet, navigate to the `$TEMP/WEB-INF/classes` directory and do the following:

- In a text editor, open `oscache.properties` (located in the `$TEMP/WEB-INF/classes` directory), and find the `cache.cluster.multicast.ip` property.
- Uncomment `cache.cluster.multicast.ip` by removing # at the start of the line. Each non-member instance requires a unique `cache.cluster.multicast.ip` value.
- Uncomment the following line by removing the # at the start of the line.

```
cache.event.listeners=com.opensymphony.oscache.plugins.clustersupport.J
avaGroupsBroadcastingListener,com.opensymphony.oscache.extra.CacheMapAc
cessEventListenerImpl
```

14. If you are using OIA Web Services, uncomment its configuration.

See the *API Guide for Oracle Identity Analytics*, "Enabling Web Services."

15. Repackage the `.war` file in the `$TEMP` directory if changes are made. Type the following commands in the test environment:

```
cd $TEMP
```

```
jar cvfM $RBACXWAR/rbacx.war .
```

16. Uninstall Oracle Identity Analytics from the application server.

17. Deploy the rebuilt Oracle Identity Analytics `.war` file on the target application server.

Refer to Chapter 4, "Deploying Oracle Identity Analytics" for instructions about Oracle Identity Analytics deployment.

18. Start the application server.

19. Go to the following address in a browser:

`https://`*Hostname*`:`*Port-Number*`/rbacx/welcome.action`

When the Welcome screen appears, enter your rbacxadmin user name and password and verify that the installation is successful.

20. Open the `rbacx.log` file and check for errors.

    The installation is successful if a message similar to the following appears in the `rbacx.log` file:

    ```
    Oracle Identity Analytics(build: 11.1.1.5.0.201107nn_nn_nnnn) Started
    ```

8.8 Complete the Post-Upgrade Steps in a Test Environment

The following items should be completed after the upgrade process completes, but before you start testing the deployment.

8.8.1 Migrate the Role Provisioning and Identity Audit Rules in the Test Environment

To migrate Role Provisioning and Identity Audit rules, open `scheduling-context.xml` using a text editor and find bean ID `quartzSchedulerFactoryBean`.

1. Stop the Oracle Identity Analytics application if it is running.
2. Locate the `jobDetails` list and uncomment `rmeRuleMigrationJob` and `identityAuditDataMigrationJob`.
3. Locate the `triggers` list and uncomment `rmeRuleMigrationJobTrigger` and `identityAuditDataMigrationTrigger`.
4. Start the Oracle Identity Analytics application.

   The Role Provisioning and Identity Audit rules migration job runs.
5. After a successful run, disable the migration job.
   a. Stop the Oracle Identity Analytics application.
   b. Navigate to the `rbacx.war` directory exploded by the application server and locate `scheduling-context.xml` under `rbacx/WEB-INF` for UNIX and under `rbacx\WEB-INF` for Windows.
   c. Open `scheduling-context.xml` using a text editor and find bean ID `quartzSchedulerFactoryBean`.
   d. Locate the `jobDetails` list and comment out `rmeRuleMigrationJob` and `identityAuditDataMigrationJob`.
   e. Locate the `triggers` list and comment out `rmeRuleMigrationJobTrigger` and `identityAuditDataMigrationTrigger`.
   f. Start the Oracle Identity Analytics application.

8.8.2 Migrate Incomplete Identity Certification Jobs in the Test Environment

If your test environment includes incomplete certifications, enable the `idcMigrationJob`. This job updates active certification data to be compatible with Oracle Identity Analytics version 11.1.1.5. This job only needs to run successfully one time in your test environment, after which it can be disabled.

8.8.2.1 To Enable the Identity Certification Migration Job

1. Stop the Oracle Identity Analytics application if it is running.
2. Navigate to *$RBACX_HOME*`/WEB-INF/` and open `scheduling-context.xml` in a text editor.
3. Uncomment the following two entries and save your changes:

- `<!--ref bean="idcMigrationJob"/-->`
- `<!--ref bean="idcMigrationTrigger"/-->`

4. To monitor the upgrade job's output, open *$RBACX_HOME*`/WEB-INF/log4j.properties`, add the following line, and save your changes:

   ```
   log4j.logger.com.vaau.rbacx.scheduling.executor.idc.IDCMigrationExecutor=DEBUG
   ```

5. Start the Oracle Identity Analytics application.

   The Identity Certification Migration job runs.

6. Open the OIA System Log and verify that the job completed successfully. The OIA System Log is located here: *$RBACX_HOME*`/logs/rbacx.log`

7. After a successful run, disable the Identity Certification Migration job:

   a. Stop the Oracle Identity Analytics application.

   b. Comment out the `idcMigrationJob` and `idcMigrationTrigger` entries and save your changes.

   c. Start the Oracle Identity Analytics application.

8.8.3 Restore Customization After the Upgrade

Set up the customized repository objects to restore application customizations.

8.9 Test the Upgrade Deployment Package

Testing is a crucial step before deploying the upgraded image to the production environment.

8.9.1 To Develop and Execute a Test Plan

Create and execute an effective test plan for the development package. A generic test plan should include the following items.

1. Introduction
   - Description of this document
   - Related documents
   - Schedule and milestones
2. Resource requirements
   - Hardware
   - Software (test tools)
3. Features to test / test approach
   - New features testing
   - Regression testing
4. Features not to test
5. Test deliverables
6. Dependencies / risk

8.9.2 To Record All Changes

It is important to document all changes that occur during the test upgrade. Consider documenting the following during the test upgrade process:

- List the version control system in use
- Verify that all existing customizations are tagged and stored in the version
- Control system
- Check all new customizations after completing the test upgrade cycle
- Create an image that consists of all the changed objects that Oracle Identity Analytics stored in the test system during the test upgrade. Deploy this image into production after upgrading the production environment.

9

Upgrading Oracle Identity Analytics in a Production Environment

This chapter contains the following sections:

9.1 Developing a Production Rollout Plan

To develop a successful production rollout plan, incorporate the information gathered while upgrading the test environment based on the following guidelines:

1. Create and document the production deployment image. Import any upgraded objects and files (not the entire environment) from the test environment.

   Create a deployment image that is compatible with the development process.

2. Create workflows and e-mail templates.

3. Document any amendments to the plan, including special procedures that apply to the production environment.

   For example:

   - Scheduling an outage for the application
   - Scheduling database administrator support
   - Notifying users before taking the system offline
   - Shutting down specific resources, processes, or applications that are used in production

9.2 Before Beginning the Upgrade Process

Before starting to upgrade Oracle Identity Analytics in a production environment, create a backup of the production data.

9.2.1 To Back Up Production Data

1. Stop all processes and all client access, and shut down Oracle Identity Analytics.

2. Take a baseline snapshot of the file system objects, the operating system, the JDK software, the repository, and the Web applications.
3. Clean up any unnecessary files and hotfixes.

9.3 Deploying the Upgrade in a Production Environment

This section provides instructions for deploying the Oracle Identity Analytics upgrade in a production environment.

9.3.1 To Deploy the Upgrade

1. Deploy from the packaged development environment.
2. Import any modifications based on differences between the test and production environments.
3. Migrate a copy of `jdbc.properties` that is suitable for a production environment.
4. Upon successfully deploying Oracle Identity Analytics to the application server, go to the following address in a browser:

   `https://`*Hostname*`:`*Port-Number*`/rbacx/welcome.action`

   When the Welcome screen appears, enter your rbacxadmin user name and password and verify that the installation is successful.
5. Open the `rbacx.log` file and check for errors.

   The installation is successful if a message similar to the following appears in the `rbacx.log` file:

   ```
   Oracle Identity Analytics (build: 11.1.1.5.0.201107_nn_nnnn) Started
   ```

9.4 Completing the Post-Upgrade Steps in a Production Environment

The following items should be completed after the upgrade process completes, but before you make the application available to users.

9.4.1 Migrate the Role Provisioning and Identity Audit Rules in the Production Environment

To migrate Role Provisioning and Identity Audit rules, open `scheduling-context.xml` using a text editor and find bean ID `quartzSchedulerFactoryBean`.

1. Stop the Oracle Identity Analytics application if it is running.
2. Locate the `jobDetails` list and uncomment `rmeRuleMigrationJob` and `identityAuditDataMigrationJob`.
3. Locate the `triggers` list and uncomment `rmeRuleMigrationJobTrigger` and `identityAuditDataMigrationTrigger`.
4. Start the Oracle Identity Analytics application.

   The Role Provisioning and Identity Audit rules migration job runs.
5. After a successful run, disable the migration job.
   a. Stop the Oracle Identity Analytics application.

b. Navigate to the `rbacx.war` directory exploded by the application server and locate `scheduling-context.xml` under `rbacx/WEB-INF` for UNIX and under `rbacx\WEB-INF` for Windows.

c. Open `scheduling-context.xml` using a text editor and find bean ID `quartzSchedulerFactoryBean`.

d. Locate the `jobDetails` list and comment out `rmeRuleMigrationJob` and `identityAuditDataMigrationJob`.

e. Locate the `triggers` list and comment out `rmeRuleMigrationJobTrigger` and `identityAuditDataMigrationTrigger`.

f. Start the Oracle Identity Analytics application.

9.4.2 Migrate Incomplete Identity Certification Jobs in the Production Environment

If your production environment includes incomplete certifications, enable the `idcMigrationJob` now. This job updates active certification data to be compatible with Oracle Identity Analytics version 11.1.1.5. This job only needs to run successfully one time in your production environment, after which it can be disabled.

9.4.2.1 To Enable the Identity Certification Migration Job

1. Stop the Oracle Identity Analytics application if it is running.
2. Navigate to *$RBACX_HOME*`/WEB-INF/` and open `scheduling-context.xml` in a text editor.
3. Uncomment the following two entries and save your changes:
   - `<!--ref bean="idcMigrationJob"/-->`
   - `<!--ref bean="idcMigrationTrigger"/-->`
4. To monitor the upgrade job's output, open *$RBACX_HOME*`/WEB-INF/log4j.properties`, add the following line, and save your changes:

   ```
   log4j.logger.com.vaau.rbacx.scheduling.executor.idc.IDCMigrationExecutor=DEBUG
   ```

5. Start the Oracle Identity Analytics application.

   The Identity Certification Migration job runs.

6. Open the OIA System Log and verify that the job completed successfully.

   The OIA System Log is located here: *$RBACX_HOME*`/logs/rbacx.log`

7. After a successful run, disable the Identity Certification Migration job:
   a. Stop the Oracle Identity Analytics application.

   b. Comment out the `idcMigrationJob` and `idcMigrationTrigger` entries and save your changes.

   c. Start the Oracle Identity Analytics application.