Oracle® Role Manager

Installation Guide

Release 10*g* (10.1.4.2)

E14608-04

February 2010

ORACLE®

Oracle Role Manager Installation Guide Release 10*g* (10.1.4.2)

E14608-04

Contents

5   Configuring IBM WebSphere

9 Upgrading Oracle Role Manager

A Manual Configuration of WebLogic Server

Index

List of Tables

Preface

Oracle Role Manager Installation Guide explains how to prepare for, install, upgrade, and configure Oracle Role Manager (Role Manager). It provides specific instructions for the operating system and Oracle software technology components that Role Manager requires.

Audience

This document is intended for Oracle database administrators (DBAs) and system administrators, and those who are involved in the installation of Oracle Role Manager and its related components.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at http://www.oracle.com/accessibility/.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Deaf/Hard of Hearing Access to Oracle Support Services

To reach Oracle Support Services, use a telecommunications relay service (TRS) to call Oracle Support at 1.800.223.1711. An Oracle Support Services engineer will handle technical issues and provide customer support according to the Oracle service request process. Information about TRS is available at

`http://www.fcc.gov/cgb/consumerfacts/trs.html`, and a list of phone numbers is available at `http://www.fcc.gov/cgb/dro/trsphonebk.html`.

Related Documents

For more information, refer to the following documents:

- *Oracle Role Manager Release Notes*
- *Oracle Role Manager Administrator's Guide*
- *Oracle Role Manager User's Guide*
- *Oracle Role Manager Developer's Guide*
- *Oracle Role Manager Java API Reference*
- *Oracle Role Manager Integration 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.

1

Overview of Oracle Role Manager Installation

Oracle Role Manager (Role Manager) is an enterprise-class application for managing business and organizational relationships, roles and entitlements. An application for role lifecycle management and an authoritative repository for roles across the IT infrastructure, it drives automation of role-based provisioning and access control. Role Manager is a component of Oracle's Identity Management and Oracle Fusion Middleware.

This chapter provides an overview of the Role Manager installation process. This chapter includes the following sections:

- Role Manager Installation Methods
- Role Manager Installation
- Installation Considerations
- About the Single Sign-On Configuration with Oracle Role Manager

1.1 Role Manager Installation Methods

You can choose different installation methods to install Role Manager, as follows:

- Interactive Installation Methods
- Silent Installations

1.1.1 Interactive Installation Methods

When you use the interactive method to install Role Manager, Oracle Universal Installer displays a series of screens that enable you to specify all of the required information to install the Role Manager software.

1.1.2 Silent Installations

This installation method is for experienced users. If you are installing Role Manager for the first time, it is recommended that you run the Oracle Universal Installer using the interactive installation method as described in "Installing Oracle Role Manager" on page 3-1.

Role Manager provides a response file template for installation (`orm.rsp`). The response template file can be found in the <ORM_installation-Media>Disk1/stage/Response directory on the Role Manager installation media.

When you start Oracle Universal Installer and specify a response file, you can automate all of the Role Manager installation. These automated installation methods are useful if you need to perform multiple installations on similarly configured systems.

Oracle Universal Installer runs in silent mode if you use a response file that specifies all required information. None of the Oracle Universal Installer screens are displayed, and the logs are created under the oraInventory location, similar to interactive mode installation.

Prepare the response file by entering values for all parameters that are missing, and then save the file. Do not edit any values in the second part of either response file.

See "Performing a Silent Installation Using a Response File" on page 3-7 for information about performing an Role Manager silent installation.

1.2 Role Manager Installation

The Role Manager installation consists of two options:

- Install Software Only–This option provides the opportunity to copy the software onto the file system and then later load the data model after customizations are put into place. This is normally done after being familiar with the product in its standard form and having identified all of the modeling changes for your business needs.

- Install Software and Configure–This is the recommended way to install Oracle Manager. This option requires that two empty database schema/users are already created, and requires connectivity to database to load the data model for a standard installation.

After you check the requirements described in "Installation Considerations" on page 1-2, the general steps to install Role Manager include these tasks:

1. Run Oracle Universal Installer to perform Role Manager installation and then upgrade the existing configuration.

2. Configure your application server with Role Manager.

3. Load the sample data for Role Manager.

1.3 Installation Considerations

This section contains information that you should consider before deciding how to install this product. It includes the following topics:

- Hardware and Software Considerations
- Multiple Oracle Homes

1.3.1 Hardware and Software Considerations

The platform-specific hardware and software requirements included in this installation guide were current at the time this guide was published. However, because new platforms and operating system versions might be certified after this guide is published, review the certification matrix on the Oracle*MetaLink* Web site for the most up-to-date list of certified hardware platforms and operating system versions. For example, the Oracle*MetaLink* Web site is available at:

http://metalink.oracle.com

If you do not have a current Oracle Support Services contract, then for example, you can access the same information at:

http://www.oracle.com/technology/support/metalink/content.html

Refer to the *Oracle Role Manager Release Notes* for detailed system requirements and supported platforms.

1.3.2 Multiple Oracle Homes

This product supports multiple Oracle homes. You can install this release of the software more than once on the same system, in different Oracle home directories.

1.4 About the Single Sign-On Configuration with Oracle Role Manager

The configuration of Oracle Access Manager with Oracle Role Manager provides a secure web-based infrastructure for role management for all customer applications and processes. Oracle Access Manager integrates identity and access management across Oracle Role Manager, enterprise resources, and other domains deployed on eBusiness networks. Oracle Access Manager provides the foundation for managing the identities of customers, partners, and employees across internet applications. These user identities are combined with security policies for protected web interaction.

For more information about Oracle Role Manager single sign-on, refer to *Oracle Role Manager Administrator's Guide*.

2

Oracle Role Manager Installation Requirements

This chapter describes Oracle Role Manager (Role Manager) installation requirements. This chapter includes the following sections:

- Host Requirements for Role Manager Components
- Planning Your Installation

2.1 Host Requirements for Role Manager Components

You must obtain enterprise versions of application server software and database software complete with valid licenses. Role Manager does not include the application server or database software.

The Role Manager installation program may conflict with other installed applications, utilities, or drivers. Try to remove all non-essential software and drivers from the installation computer before loading Role Manager.

> **Important:** Always check the *Oracle Role Manager Release Notes* for the requirements and supported configurations specific to each version. The information in this guide applies to the Oracle Role Manager 10.1.x versions.

2.1.1 Role Manager Application Server Host Requirements

Refer to the *Oracle Role Manager Release Notes* for the specific application server host requirements.

2.1.2 Database Server Host Requirements

Refer to the Oracle Database documentation for the specific database host requirements.

2.2 Planning Your Installation

Before installing Role Manager, you must read "Host Requirements for Role Manager Components" on page 2-1 and "Installation Worksheet" on page 2-2 to help plan your installation.

Because the Database Administrator (DBA), System Administrator, and IT Developer typically handle tasks specific to their specific areas of expertise, you should share Role Manager installation information among your team members.

2.2.1 Installation Worksheet

The Installation Worksheet table helps you identify configuration attributes you need before starting the Role Manager installation. Print this worksheet and use it to take notes as you go through your installation. Use the User Selection column to fill in information specific to your installation:

Table 2–1   Installation Worksheet

Item	User Selection
Base directory for installing Role Manager.	
Name or IP address of the computer where the Role Manager database is installed.	
TCP listener port number for the database..	
Service name of the database for your installation.	
User name and password of the SYSTEM user account for access to the database.	
Name or IP address of the application server computer.	
TCP listener port number for the application server.	

3

Installing Oracle Role Manager

This chapter includes the major steps required to install Oracle Role Manager (Role Manager).

This chapter includes the following sections:

- Before You Start
- Database Setup
- Installing Role Manager
- Performing a Silent Installation Using a Response File
- Troubleshooting Configuration Assistant

3.1 Before You Start

Before you begin the Role Manager installation, you must create a Role Manager database user (owner) and Role Manager application user.

Before performing the procedures in this chapter, ensure the following:

- The installation computer has network access to the database server host.
- You have the necessary information from the Installation Worksheet on page 2-2.
- For UNIX-based systems—It is recommended that you create a special user account, such as `orm`, and set its home directory to the directory you plan to use for installation. You may want to have all dependent applications (such as JBoss and WebSphere) participate in the same group.

3.2 Database Setup

Before installing Role Manager, you need to create the database owner and application user schemas on the database used for Role Manager. Database owner is the user with permissions to change the schema, but the application user does not have permissions to change the schema. It is recommended that you use the scripts provided on the installation media following the procedures in this section. These steps, described in this section, are normally performed by a database administrator on the Oracle database host.

Before continuing, ensure that you have met the following requirements:

- The Oracle service and TNSListener are running.
- You have the Oracle `SYSTEM` account user name and password.
- You have the Oracle service name (instance).

■ Optional: You have OS-level permission to edit the init.ora file to configure the Oracle rollback segments created as part of the Role Manager installation.

3.2.1 Creating the Role Manager Tablespaces

Before the Role Manager data model can be deployed, either by the Role Manager installer or manually at the command line, the tablespaces for Role Manager must exist.

Your policies may require that the scripts referenced in this section be run directly on the Oracle database server by a database administrator.

The provided tablespace creation script needs to be modified if your organization requires different extent and growth options. If you instead prefer to use your own tablespace creation script, the Role Manager user creation scripts expect three tablespaces, one for DATA pages named ORM_DATA, one for INDEX pages named ORM_INDEX, and one for TEMP space.

Data, Index, and Temporary Tablespaces

Data pages are files that are contained by each tablespace in an Oracle database. A data file can be associated with only one tablespace and only one database. Before you can create a tablespace, you must create a database to contain it. The primary tablespace in any database is the SYSTEM tablespace. The SYSTEM tablespace always contains the data dictionary tables for the entire database.

Index pages are either stored in the user's default tablespace or in the tablespace specified in the CREATE INDEX statement.

Temporary table space is used for temporary storage of data. A temporary tablespace can be assigned to users with CREATE USER or ALTER USER statement and can be shared by multiple users.

The directory path to locate data, index, and temporary tablespaces is `$ORACLE_DB_HOME\oradata\$ORACLE_SID\`

For example:

`C:\oracle\product\10.1.4\oradata\orcl\`

For more information about table spaces, visit the Oracle Web site at

http://download.oracle.com/docs/cd/B19306_01/server.102/b14231/tspaces.htm#sth

To create the Role Manager tablespaces:

1. From the installation media, copy the create-tablespace.sql script from the samples/sqlscripts/oracle directory to a temporary location.

2. Optionally, in the create-tablespace.sql file, edit the path to the log directory you want to use to save messages created by this script. For example:

   ```
   set echo on;
   spool <new_log_path>/create-tablespace.log
   ```

3. Optionally, in the same file, modify the tablespace definitions to meet your specific configuration requirements.

> **Note:** If you modify the actual tablespace names in the
> create-tablespace.sql script in any way, you must also modify the
> `create-schema-owner.sql` and `create-app-user.sql` scripts,
> as appropriate, to accommodate the changed tablespace names.

4. Using sqlplus or a similar utility, run the create-tablespace.sql script to create the tablespaces for Role Manager database table data, index, temporary data, and rollback segments. For example:

```
sqlplus system/<system_pw><db_service> create-tablespace <data_dir>/ORM_DATA
<index_dir>/ORM_INDEX <temp_dir>/ORM_TEMP
<undo_dir/ORM_UNDO
```

> **Note:** The data and index table spaces must be named ORM_DATA
> and ORM_INDEX respectively.

5. Review the contents of the create-tablespace.log file to verify there were no error messages.

   If there are errors, ensure that the Oracle database paths you specified in the command are valid.

6. Optionally, configure rollback segments.

   Rollback segments are used to save data in transactions before data is committed to the database. To bring the rollback segments online automatically every time the Oracle instance is restarted, you must update the init.ora file by adding the following line

```
ROLLBACK_SEGMENTS = (ORM_1, ORM_2, ORM_3, ORM_4)
```

3.2.2 Creating the Role Manager Users

Before creating the Role Manager users, ensure that the tablespaces used for Role Manager have been created.

To create the Role Manager database users:

1. From the installation media, copy the following scripts from the samples/sqlscripts/oracle directory to a temporary location:

```
create-app-user.sql
create-schema-owner.sql
```

2. As the Oracle SYSTEM user, using sqlplus or a similar utility, create the Role Manager database owner by running the create-schema-owner.sql script as follows:

```
sqlplus system/[system_pw][db_service] create-schema-owner [owner_username]
```

   Enter the password at the prompt and make note of username and password values, because you need them when running the Role Manager installer.

   If you have changed tablespace names, ensure to update create-schema-owner.sql to use the same names as those defined in the create-tablespace.sql file.

> **Note:** Errors in the console in the form "does not exist" are normal the first time this script and the following script are run.

3. Create the application user for Role Manager by running the create-app-user.sql script as follows:

```
sqlplus system/[system_pw][db_service] create-app-user [app_username]
```

Enter the password at the prompt and make note of username and password values, because you need them when running the Role Manager installer. If you have changed tablespace names, ensure to update create-app-user.sql to use the same names as those defined in the create-tablespace.sql file.

Role Manager uses two schema/users, one as the owner and the other as the user of the application. This is done for security reasons, ensuring that schema changes can be made only by the owner. As part of the configuration process, the installer creates all of the synonyms required so that the Role Manager application user can invoke data changes.

3.3 Installing Role Manager

In this part of the installation, you launch the installer and configure Role Manager with your environment.

To perform the Role Manager installation:

1. Ensure that the drive or directory containing the Role Manager installation media is accessible from the installation computer.

2. Run Oracle Universal Installer (OUI) to install Role Manager.

   For Windows:

   - Locate the directory containing the Role Manager installation files for Windows.

   - Double-click setup.exe.

   For UNIX:

   - Change directory to the location containing the Role Manager installation files for your operating system.

   - As the software owner account that owns the current *ORACLE_HOME* environment, start the installer with the following command:

     ```
     ./runInstaller.sh
     ```

   > **Note:** If you do not know the account to use, contact your system administrator or DBA for information specific to your environment.

3. On the Welcome page, click **Next**.

4. On the Specify File Location page:

   a. Ensure that the source path is pointing to the Role Manager installation media, for example:

   ```
   C:\stage\products.xml
   ```

b. If you already have any oracle products on the installation computer, type a new unique identifier that Oracle can use for inventory purposes, such as `orm`.

c. Change the destination path to the location on the file system where you want to install the Role Manager software.

> **Note:** The destination must be an empty directory. The installer creates the directory you specify if it does not already exist.

5. Choose one of the following installation options, then click **Next** to continue.

   ■ Install Software and Configure

   This option installs the Role Manager software and then configures the database with the primordial data model. It also provides additional options to include the standard data model used by the Role Manager UI. If you do not yet have a customized data model, select this option.

   ■ Install Software Only

   This option copies the Role Manager files to the specified installation location. It does not load any configuration or deploy the models required to use the Role Manager UI. This option is intended for manual deployments of customized models. Choosing this selection skips to the Summary page of the installer. For information about deploying the customized data model, refer to "Manual Data Model Deployment" on page 7-2.

6. Select the configuration options you want, then click **Next** to continue.

   ■ Primordial Data Model

   The core model used by the Role Manager system components.

   ■ Standard Data Model

   This is required to load the standard configurations to Oracle Role Manager system.

7. Set the database connection values in the Database Connect String as follows:

   a. For a single database host instance, provide the connection string as follows:

   ```
   <Database-server host name>:<database server port number>:<Database service
   name>
   ```

   b. For a real application cluster (RAC) database instance, provide the connection string as follows:

   ```
   <Database-server instance1 host name>:<database server instance1 port
   number>^<Databaseserver instance2 host name>:<database server instance2
   port number>@<Database service name>
   ```

   For example:

   ```
   host1-vip:1521^host2-vip:1522@db-service
   ```

8. Set the values needed to configure the Role Manager database tables as described below.

   a. In the **Database Owner** field, type the name to use for the Role Manager owner schema/user.

b. In the **Database Owner Password** field, type the password to use for the Role Manager owner schema/user.

c. In the **Application User** field, type the name to use for the Role Manager application user schema.

d. In the **Application User Password** field, type the password to use for the Role Manager application user schema.

e. Click **Next** to continue.

9. Create the administrative account for Role Manager as follows:

a. Type a name to use as the Role Manager Administrator user ID.

b. Type a password for the Role Manager Administrator.

c. Retype the password in the Confirm Password field, then click **Next** to continue.

Oracle Universal Installer checks the system to verify that it is configured correctly to run Oracle software. If you have completed all of the preinstallation steps in this guide, all of the checks should pass.

If a check fails, then review the cause of the failure listed for that check on the screen. If possible, rectify the problem and rerun the check. Alternatively, if you are satisfied that your system meets the requirements, then you can select the check box for the failed check to manually verify the requirement.

10. On the installation Summary page, review the installation summary information. After reviewing this installation information, click **Install** to begin the installation procedure.

11. If you encounter problems during installation, then examine the Oracle Universal Installer actions recorded in the installation log file. The log file can be found in the cfgtoollogs/oui directory, in the following location:

For Windows:

```
%SYSTEMDRIVE%\Program Files\Oracle\Inventory\oraInventory
```

For UNIX:

If your computer already has the oraInst.loc file at the "/etc" location, then oraInventory is created at that location. This is depicted in the following example:

```
cat /etc/oraInst.loc
inventory_loc=/home/spatra/oraInventory123
```

If your computer is not having the oraInst.loc file at the "/etc" location, then oraInventory is created by default at $HOME/oraInventory, where $HOME is the user home.

> **Note:** When runInstaller.sh (linux or aix) or setup.exe (windows) is executed, it displays the log location.

12. After the installation completes, click **Exit**, then click **Yes** to confirm.

> **Note:** In case of installing and configuring Oracle Role Manager software, the users/schemas must already exist and name and password values must match what was used when they were created. Refer to "Creating the Role Manager Users" for information.

3.4 Performing a Silent Installation Using a Response File

Follow these brief steps to perform a silent installation using a response file:

1. Ensure that all prerequisites are met for the installation of Role Manager.

2. Prepare the Role Manager response file. A template response file can be found with the Role Manager installation media at stage/Response/orm.rsp.

   Alternatively, you can run the installer in a record mode to save your inputs to a file, which you can use later as a response file. Use the following command to run the installer in a record mode:

   ```
   setup.exe -record -destinationFile newResponseFile
   ```

   The version of the response file used is 2.2.1.0.0. Prepare the response file by entering values in the file for all parameters, then save the file. Table 3–1 describes the parameters of the response file with its sample values.

Table 3–1   Parameters of the Response File

Parameter	Description	Sample Value
UNIX_GROUP_NAME	Enter the UNIX group name to be set for the inventory directory. **Note:** This is valid only for UNIX platforms.	g533
FROM_LOCATION	Enter the source path to locate the installation media, products.xml file. This file is present in the location where setup.exe file is present.	C:\Softwares\build2\DAILY\Disk1\stage\products.xml
ORACLE_HOME	Enter the valid path of the ORACLE_HOME.	C:\OHOME1
ORACLE_HOME_NAME	Enter the Oracle_Home_Name, where folders and services are created.	OHOME1
TOPLEVEL_COMPONENT	This field holds the details of the top-level component to be installed in the current session. **Note:** You need not change the default value of this parameter.	("oracle.orm.top", "10.1.4.0")
DEINSTALL_LIST	List of components to be de-installed. **Note:** You must enter a value for this parameter only if you are de-installing any component using response file.	oracle.orm.top
SELECTED_LANGUAGES	The language in which the components are installed. **Note:** You need not change the default value of this parameter.	en
INSTALL_TYPE	Installation type of the component. **Note:** You need not change the default value of this parameter.	Oracle Role Manager

Table 3–1   (Cont.) Parameters of the Response File

Parameter	Description	Sample Value
SZL_RepositoryUserInput	Enter the repository details of the user.	"DB-USERNAME" "DB-PASSWORD"
szl_ORMAdminInput	Enter the details of Oracle Role Manager Administrator.	"ADMIN-USERNAME" "ADMIN-PASSWORD"
StartupProcesses	Indicates the following configuration options: ■ Primordial Data Model - The core model used by the Role Manager system components. ■ Standard Data Model - to load the standard configurations to Oracle Role Manager system. **Note:** You need not change the default value of this parameter.	Primordial Data Model Configuration
n_Install Type	Enter a valid number to indicate the installation type. You must enter "0" for Install and Configure and "1" for Install Only option.	0

3. Invoke Oracle Universal Installer using the following options:

    For UNIX:

    ```
    ./runInstaller.sh -silent -responseFile <path_to_rsp>
    ```

    For Windows:

    ```
    setup.exe -silent -responseFile <path_to_rsp>
    ```

    In this example:

    `Path_to_rsp` identifies the full path of the response file.

    `-silent` runs Oracle Universal Installer in silent mode and suppresses the Welcome window.

    For general information about these options and about how to complete an installation using these response files, see the platform specific Oracle Database installation guides and *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Installation Guide for Linux* and "Installing Oracle Products" in *Oracle Universal Installer and OPatch User's Guide* for more information about installing and using response files.

    > **Note:**   In case of installing and configuring Oracle Role Manager software, the users/schemas must already exist and name and password values must match what was used when they were created. Refer to "Creating the Role Manager Users" for information.

3.5 Troubleshooting Configuration Assistant

If Oracle Role Manager configuration assistant failure occurs when running configuration assistant execution commands on the command line, then re-run the configuration assistant execution command. The configToolCommands file will be

generated under *ORM_HOME*/cfgtoollogs directory. You can use this generated script file to execute the failed configuration assistant.

4

Configuring WebLogic Server

This chapter includes the steps required to configure the application server to run the Oracle Role Manager (Role Manager) server and Web application.

This chapter includes the following sections:

- Before You Configure the Application Server
- Configuring Oracle WebLogic Server in a Nonclustered Mode
- Configuring Oracle WebLogic Server in a Clustered Mode

4.1 Before You Configure the Application Server

Role Manager is intended to be deployed on only one server platform per installation. The server platform can be one of the following:

- Oracle WebLogic Server
- JBoss
- IBM WebSphere

The procedures in this chapter assume the following:

- The application server system has network access to the database server host.
- The Role Manager software is accessible by the application server system.
- You know the application server's listener port and host name.
- If running on WebSphere, a server, cell, and node to use for Role Manager have been created and configured.

While configuring the application servers, you may require to modify the following files:

- webui.war file - This file is used for customizing Oracle Role Manager to the user's requirement.

  > **Note:** For WebSphere application server, webui.ear file is used and not webui.war file.

- `orm-ds.xml`file - This file is used to configure the data source information. Refer Step 5 of the "Configuring JBoss Server in a Nonclustered Mode" on page 6-1 for information about how to modify the data source information in the `orm-ds.xml` file.

4.2 Configuring Oracle WebLogic Server in a Nonclustered Mode

You can configure the WebLogic server either manually or automatically, following an extension template. This section provides the information about automated configuration of WebLogic server. If you want to configure the WebLogic server manually, refer to Appendix A.

> **Note:** Before you run the automated configuration of WebLogic server, ensure that the WebLogic server is installed and the node manager is up and running. You can start the node manager by running the
> *BEA_HOME*/wlserver_10.3/server/bin/startNodeManager.sh script for Linux.
>
> *BEA_HOME*/wlserver_10.3/server/bin/startNodeManager.cmd script for Microsoft Windows.

You must configure WebLogic server in SSL mode to operate in a secure environment. For information about configuring SSL for WebLogic server, refer to the following URL:

http://e-docs.bea.com/wls/docs103/secmanage/ssl.html

This section includes the following topics:

- Configuring WebLogic Server Based On Template
- Setting Up Commons Logging
- Configuring Administrative Console
- Deploying Role Manager

4.2.1 Configuring WebLogic Server Based On Template

To perform the template based configuration of WebLogic server:

1. Run the configuration wizard in WebLogic server directory:

   BEA_HOME\wlserver_10.3\common\bin\config.exe or config.sh

2. When the configuration wizard is displayed, select **Create a new WebLogic Domain** and click **Next**.

3. Select **Base this domain on an existing template**, go to the following path and click **Next**:

   ORM_HOME/weblogic/templates/10.3/orm_createdomain_template_103.jar

4. In the Specify Domain Name and Location window, type domain name and click **Next**.

5. In the Configure Administrator Username and Password window, type **User Name** and **Password** and click **Next**.

6. Configure server start mode and JDK by performing the following steps:

   a. On the left side of the window, select **Production Mode**.

   b. On the right side of the window, select appropriate JDK and click **Next**.

7. In the Configure Data Sources window, for both the ORM Data Source and the ORM XA Data Source, type the DBMS details such as User Name, Password, DBMS/Service, Host, and Port and click **Next**.

> **Note:** If you are using RAC database, provide the following string while creating the data source, by substituting the values for hostname, port, and service name of your instance:
>
> ```
> jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP
> )(HOST=host1-vip)(PORT=1521))(ADDRESS=(PROTOCOL=TCP)(HOST=host2-vip
> )(PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=orcl.us.oracle.com)))
> ```

8. In the Test JDBS Data Sources window, ensure that you click **Test Connections** to test the data connection for each datasource and then click **Next**.

9. In the Select Optional Configuration window, select all check boxes and click **Next**.

10. In the Configure Administration Server window, set the Listen Address to a value appropriate to your setup such as, LocalHost, IP Address, DNS equivalent and Listen Port and then click **Next**.

11. In the Configure Managed Servers window, set the Listen Address to a value appropriate to your setup such as, LocalHost, IP Address, DNS equivalent and Listen Port and then click **Next**.

12. In the Configure Clusters window, click **Next**.

13. In the Configure Machines window:

- for UNIX machine, click the **UNIX machine** tab, the Oracle Role Manager machine name is auto populated. Click **Next**.

- for Windows machine, click the **UNIX machine** tab and delete the existing Oracle Role Manager machine name, **ormMachine.** On the **Machine** tab, click **Add** and then type Oracle Role Manager machine name, ORM_Machine, Node manager listen address, and Node manager listen port, and then click **Next**.

14. In the Assign Servers to Machines window, ensure that ORMServer is listed under ormMachine on the right window and then click **Next**.

> **Note:** For Microsoft window, select ORM_Machine on the right window.

15. In the Target Services to Clusters or Servers window, click **Next**.

16. In the Configure RDBMS Security Store Database window, click **Next**.

17. In the Configuration Summary window, verify the details and click **Create**. The domain is created and the following are configured automatically:

a. Admin server is created.

b. Oracle Role Manager Managed server, ORMServer is created.

c. Non-XA Data Source, ORM Data Source is created.

d. XA Data Source, ORM XA Data Source is created.

e. JMS Server, ORM JMSServer is created.

    f. JMS Module, ORM JMSModule is created.

    g. Subdeployment, ORM JMSSubdeployment is created.

    h. Topic, ORM NotificationTopic is created.

    i. Queues, ORM FinisherQueue, ORM LoaderQueue, and ORM IncomingEventQueue are created.

    j. Connection Factory such as ORM Connection Factory and Finalization ORM Connection Factory are created.

18. Click **Done**.

4.2.2 Setting Up Commons Logging

Set up the Commons Logging by performing the steps described in "Setting Up Commons Logging" on page A-4.

> **Note:** Logging will take effect the next time WebLogic Administration server and Oracle Role Manager server are started.

4.2.3 Configuring Administrative Console

1. Start the server and log in to the WebLogic Administrative Console.

2. Set the JTA transaction timeout parameter by performing the following steps:

    a. In the domain tree, select **Services**, **JTA**.

    b. In the **Timeout Seconds** field, type `1200`.

    c. In the **Abandon Timeout Seconds** field, ensure that the value is `86400`.

    d. Click **Save**.

> **Note:** The value of Abandon Timeout Seconds must always be greater than Timeout Seconds and Stuck Thread Max Time.

3. Set the Stuck Thread Max Time parameter by performing the following steps:

    a. In the domain tree, select **Environment**, **Servers**, **ORMServer**.

    b. Click the **Tuning** tab.

    c. In the **Stuck Thread Max Time** field, type `3000`.

    d. Click **Save**.

> **Note:** The value of Stuck Thread Max Time must be at least twice that of the value of Timeout Seconds.

4. Configure the JMS Connection Factory by performing the following steps:

    a. In the domain tree, select **Services**, **Messaging**, **JMS Modules**.

    b. Click **ORM JMS Module**.

    c. Click **ORM ConnectionFactory**.

    d. Ensure that the **Default Targeting Enabled** field is selected.

    e. Click **Save**.

    f. Click the **Transactions** tab, ensure that the **XA Connection Factory Enabled** field is selected and then click **Save**.

    g. In the domain tree, select **Services**, **Messaging**, **JMS Modules.**

    h. Click **ORM JMS Module**.

    i. Click **Finalization ORM ConnectionFactory**.

    j. Ensure that the **Default Targeting Enabled** field is selected.

    k. Click **Save**.

    l. Click the **Transactions** tab, ensure that the **XA Connection Factory Enabled** field is selected and then click **Save**.

5. Configure the Redelivery Limit by performing the following steps:

    a. In the domain tree, select **Services**, **Messaging**, **JMS Modules**.

    b. Click **ORM JMS Module**.

    c. Click **ORM FinisherQueue**.

    d. Click **Delivery Failure**.

    e. Set Redelivery Limit value to 3.

    f. Click **Save**.

    g. In the domain tree, select **Services**, **Messaging**, **JMS Modules**.

    h. Click **ORM JMS Module**.

    i. Click **ORM IncomingEventQueue**.

    j. Click **Delivery Failure.**

    k. Set Redelivery Limit value to 3.

    l. Click **Save**.

    m. In the domain tree, select **Services**, **Messaging**, **JMS Modules**.

    n. Click **ORM JMS Module**.

    o. Click **ORM LoaderQueue**.

    p. click **Delivery Failure**.

    q. set Redelivery Limit value to 3.

    r. Click **Save**.

6. In the domain tree, select **Environment, Servers, ORM Server** and navigate to the **Control** tab and click **Start** to start the managed server.

> **Note:** You must start node manager before starting the managed server.

4.2.4 Deploying Role Manager

> **Note:** Perform this procedure only if you are repeating the deployment of the same Role Manager server.
>
> If this is your first installation, then skip this section and continue with "To deploy the Role Manager server application".

To deploy Role Manager:

1. Start up the Role Manager server that contains the JMS server.

2. Navigate to Services, Messaging, and JMS Modules.

3. Click **ORM JMSModule**.

4. From the resource list, select the **ORMFinisherQueue**.

5. On the Monitoring tab, select the **ORM JMSModule! ORM Finisher Queue** destination and click **Show Messages**.

6. Click **Delete** and from the list of options, select **Delete All**.

7. In the confirmation page, click **Yes**.

To deploy the Role Manager server application:

1. Select **Deployments** in the left part of the Administration Console window.

2. Click **Install** in the right part of the Administration Console window.

3. In the *ORM_HOME*, navigate to the `server.ear` file and select it. Click **Next**.

   > **Note:** The `server.ear` file can be accessed from the path *ORM_HOME*/lib/server.ear.

4. Select **Install this deployment as an application**. Click **Next**.

5. Select **ORM Server** as the target in the **Target** field. Click **Next**.

6. In the **Name** field, type the name of the Role Manager server application, for example, `ORM ServerApp`.

7. Click **Next** and in the Install Application Assistant page, click **Finish**.

8. Click **Save** to activate the changes.

   > **Note:** You must restart the ORMServer to login.

9. In the right part of the Administration Console window, under Deployments, check the State of ORM Server App. If it displays: **Prepared**, then select the checkbox next to the name of it. Click **Start** and from the list, select **Servicing All Requests**. When the status of the application is Active, you can log into the Oracle Role Manager Console.

10. To test the server installation, ensure that you can get to the Role Manager administrative console from a Web browser. For example:

    ```
    http://localhost:<port>/ormconsole
    ```

> **Note:** The default Oracle Role Manager application server port is 9001.

You should see the Home page of the Role Manager administrative console.

To deploy the Role Manager Web application:

1. Select **Deployments** in the left part of the Administration Console window.

2. Click **Install** in the right part of the Administration Console window.

3. Navigate to the `webui.war` file and select it. Click **Next**.

   > **Note:** The `webui.war` file can be accessed from the path *ORM_HOME*//webui/weblogic/10.3/webui.war.

4. Select **Install this deployment as an application**. Click **Next**.

5. In the **Target** field, select **ORMServer** as the target and then click **Next**.

6. In the **Name** field, type the name of the Web user interface application.

7. Select **Custom Roles and Policies: Use only roles and policies that are defined in the Administration Console** from the **Security** list.

8. Click **Next** and then click **Finish**.

9. In the right part of the Administration Console window, under Deployments, check the State of Webui App. If it displays: **Prepared**, then select the checkbox next to the name of it. Click **Start** and from the list, select **Servicing All Requests**. When the status of the application is Active, you can log into the Oracle Role Manager Console.

10. Test the Web application installation as follows:

    a. In a Web browser, navigate to the Role Manager Web application address. For example:

       `http://localhost:<port>/webui`

    b. Log in as the Role Manager Administrator created in "Installing Role Manager" on page 3-4.

       > **Note:** Data must be loaded into the system to expose all the functionality of the application. Refer to "Loading Sample Data" on page 7-2 for instructions.

You should see the Home page of the Role Manager Web application.

To start the server application and the Web application for WebLogic:

Starting the server application and the Web application are automatically taken care for WebLogic. If server is running in Production mode, then application will be in prepared state. To change it to active state, perform the following steps:

1. Click **Deployments** and select an application.

2. In the right part of the Administration Console window, under Deployments, check the State of ORM Server App. If it displays:

- **Active**, then the application is started and you need not perform the following steps.

- **Prepared**, then select the checkbox next to the name of it. Click **Start** and from the list, select **Servicing All Requests**. When the status of the application is Active, you can log into the Oracle Role Manager Console.

4.3 Configuring Oracle WebLogic Server in a Clustered Mode

This section explains how to deploy Oracle Role Manager in a clustered Oracle WebLogic Server environment.

This section discusses the following topics:

- About Oracle WebLogic Server Clusters
- Configuring WebLogic Clusters
- Configuring Apache Failover Proxy

4.3.1 About Oracle WebLogic Server Clusters

A clustered installation requires multiple host computers. The instructions in this chapter involve deployment and running of Oracle Role Manager on three host computers. These instructions assume that you have three computers, of which one is used to host the Web server and the other two are used for Oracle Role Manager cluster.

Table 4–1 describes the entities needed for a cluster, the computers that the entities run on, and the software required for the entities. Host computers and entities are labeled.

Table 4–1   WebLogic-Based Oracle Role Manager Cluster Host Computers

Host Computers	Entities	Software	Description
PRIMARY_NODE on ORM_Machine	WebLogic Admin Server, and WebLogic Node Manager	WebLogic Server	Administrative server for the WebLogic domain and WebLogic Node Manager
	ORM Finalization Server	Oracle Role Manager	ORM Finalization Server
	ORM_SERVER1	Oracle Role Manager	WebLogic Managed Server, part of ORM_CLUSTER
	ORM_CLUSTER	Oracle Role Manager	Name of the WebLogic cluster that hosts Oracle Role Manager (logical entity).
SECONDARY_NODE on ORM_Machine1	WebLogic Node Manager	WebLogic Server	WebLogic Node Manager
	ORM_SERVER2	Oracle Role Manager	WebLogic Managed Server 2, part of ORM_CLUSTER
	ORM_CLUSTER		Name of the WebLogic cluster that hosts Oracle Role Manager (logical entity).
THIRD_NODE on ORM_Machine2	Apache Web Server	Apache HTTP Server 2.2 with WebLogic 10.3 Apache plugin	Apache Failover Proxy

> **Caution:** Deploying an application in a clustered installation is a complex procedure. This document assumes that you have expertise in installing and running applications on an Oracle WebLogic Server cluster. This chapter provides Oracle Role Manager-specific information only. It does not cover the procedure to set up an Oracle WebLogic Server cluster. For more information about clustering, refer to Oracle WebLogic Server documentation.

4.3.2 Configuring WebLogic Clusters

The instructions mentioned in this section are for installing Role Manager in a WebLogic cluster of two machines with two Role Manager servers, one Admin Server and one finalization server. Therefore there are four servers, the PRIMARY_NODE on first machine (ORM_Machine) hosts Admin, Finalization, and Managed Server1 and the SECONDARY_NODE on second machine (ORM_Machine1) hosts Managed Server2. In addition, the data store for the queues are database-based and not file-based.

4.3.2.1 Configuring WebLogic Server in a Clustered Mode

To configure WebLogic server in a clustered mode:

1. On the primary node, complete the single instance of WebLogic server configuration, install and deploy Role Manager, load sample data, and ensure that the server is running. Refer to "Configuring WebLogic Server Based On Template" on page 4-2 for instructions on configuring the WebLogic server.

   The following are the configuration details for the primary node:

   Host: PRIMARY_NODE

   Machine: ORM_Machine

   BEA_Home: BEA_HOME

   WLS_HOME: WLS_HOME

   DOMAIN_HOME: DOMAIN_HOME

   ADMIN_PORT: ADMIN_PORT

   ORM server: ORMServer

   ORM server Port: ORMServer_Port (9001)

   Node_manager Port: NODE_MANAGER_PORT (5556, ssl)

   To control your Managed Servers remotely from the Administrative Server, you must set up and configure Node Manager on each of the remote systems hosting Managed Servers by following the instructions on the BEA e-docs page: http://e-docs.bea.com/wls/docs103/nodemgr/nodemgr_config.html

   The default location of the nodemanager.properties file is:

   BEA_HOME/wlserver_10.3/common/nodemanager

2. Install WebLogic on a secondary node.

3. Start the node manager on both the nodes and start the administrative server on primary node if it is not running.

4. Log in to the administrative server console on the primary node. For example: *http://primarynode:7001/console*.

5. Shutdown ORMServer on the primary node as follows:

   a. In the domain tree, select **Environment**, **Servers**.

   b. Click the **Control** tab.

   c. Select **ORM Server** and then click **Shutdown**.

6. Create a Machine, for example, ORM_Machine1 for the secondary node as follows:

   a. In the domain tree, select **Environment**, **Machines**.

   b. Click **New**.

   c. In the **Name** field, type ORM_Machine1.

   > **Note:** For UNIX flavors, in the **Machine OS** field, select UNIX.

   d. Click **OK**.

7. Configure the machine to access Node Manager on secondary machine as follows

   a. In the domain tree, select **Environment**, **Machines**.

   b. Click on the machine that you created, for example, ORM_Machine1.

   c. Click the **Node Manager** tab.

   d. In the **Listen Address** field, type the IP address of the secondary node and click **Save**.

8. Create a server, for example, ORMServer1 which uses the port, for example, ORMServer_Port1(7071) as follows:

   a. In the domain tree, select **Environment**, **Servers**.

   b. Click **New**.

   c. In the **Server Name** field, type ORMServer1.

   d. In the **Server Listen Port** field, type 7071.

   e. Click **Finish**.

9. Ensure that ORMServer1 is assigned to ORM_Machine1 as follows:

   a. In the domain tree, select **Environment**, **Machines**, **ORM_Machine1**, **Node Manager**.

   b. Check whether the IP address is same as the secondary node.

   c. Click the **Servers** tab and select **Add**.

   d. Choose **Select an existing server, and associate it with this machine** and from the **Select a server** list, select **ORMServer1**.

   e. Click **Finish**.

   The following are the configuration details of the secondary node:

   Host: SECONDARY_NODE

   Machine: ORM_Machine1

   BEA_Home: BEA_HOME1

WLS_HOME: WLS_HOME1

DOMAIN_HOME: No domain yet

ADMIN_PORT: No admin server on secondary node

ORM server: ORMServer1

ORM server Port: ORMServer_Port1 (7071)

Node_manager Port: NODE_MANAGER_PORT1 (5556, ssl)

10. Create cluster as follows:

   a. In the domain tree, select **Environment**, **Clusters**.

   b. Click **New**.

   c. In the **Name** field, type ORMCluster.

   d. In the Messaging Mode list, select Multicast.

   > **Note:** The default values for multicast address and multicast port can be used only if they do not conflict with any other cluster in the same domain.

   e. Click **OK**.

   f. Click **ORMCluster**.

   g. Click **Servers** tab.

   h. Click **Add** to add a server to cluster.

   i. Select the server, ORMServer and click **Finish**.

   j. Repeat the steps g to h and select the second server, ORMServer1.

4.3.2.2 Configuring JDBC Stores

1. Configure the JDBC data sources as follows:

   a. In the domain tree, select **Services**, **JDBC**, **Data Sources**.

   b. Click **ORM Data Source**.

   c. Click the **Targets** tab.

   d. Select **ORMCluster**, **All servers in the cluster**.

   e. Click **Save**.

   f. Repeat the steps a to f for ORM XA Data Source, except that in the Step c, click **ORM XA Data Source**.

2. Create a JDBC Store as follows:

   a. In the domain tree, select **Services**.

   b. Click **Persistent Stores**.

   c. Click **New** and select **Create JDBC Store** from the list.

   d. In the **Name** field, type JDBCStore.

   e. In the **Target** field, select **ORMServer**.

   f. In the **DataSource** field, select **ORM Data Source**.

    g. In the **Prefix Name** field, type `jdbcstore`.

    h. Click **OK**.

3. Create another JDBC Store as follows:

    a. In the domain tree, select **Services**.

    b. Click **Persistent Stores**.

    c. Click **New** and select **Create JDBCStore** from the list.

    d. In the **Name** field, type `JDBCStore1`.

    e. In the **Target** field, select **ORMServer1**.

    f. In the **DataSource** field, select **ORM Data Source**.

    g. In the **Prefix Name** field, type `jdbcstore1`.

    h. Click **OK**.

4.3.2.3 Configuring JMS Servers

1. Create a JMS server for the secondary node as follows:

    a. In the domain tree, select **Services**, **Messaging**.

    b. Click **JMS Servers, ORM JMSServer**.

    c. In the **Persistent Store** field, select **JDBCStore**.

    d. click **Save**.

2. Create another JMS server in the secondary node as follows:

    a. In the domain tree, select **Services**, **Messaging**.

    b. Click **JMS Servers**.

    c. Click **New**.

    d. In the **Name** field, type `ORM JMSServer1`.

    e. In the Persistent Store field, select **JDBCStore1** and then click **Next**.

    f. In the **Target** field, select **ORMServer1**.

    g. Click **Finish**.

3. Configure JMS Modules as follows:

    a. In the domain tree, select **Services**, Messaging, JMS Modules.

    b. Click **ORM JMSModule**.

    c. Click the **Targets** tab.

    d. Select **ORMCluster**, **All servers in the cluster**.

    e. Click **Save**.

4.3.2.4 Configuring a Subdeployment

1. Create a Subdeployment as follows:

    a. In the domain tree, select **Services**, **Messaging**, **JMS Modules**.

    b. Click **ORM JMSModule**.

    c. Click the **Subdeployments** tab.

   d. Click **New**.

   e. In the **Subdeployment Name** field, type `ORM JMSSubdeployment1`.

   f. Click **Next** and select **ORM JMSServer1** as the target JMS server.

   g. Click **Finish**.

2. Create a second Subdeployment as follows:

   a. In the domain tree, select **Services**, **Messaging**, **JMS Modules**.

   b. Click **ORM JMSModule**.

   c. Click the **Subdeployments** tab.

   d. Click **New**.

   e. In the **Subdeployment Name** field, type `cf-sub`.

   f. Click **Next** and select **ORMCluster**, **All servers in the cluster**.

   g. Click **Finish**.

4.3.2.5 Configuring JMS Topics and Queues

1. Create JMS Topics and Queues using Distributed Option as follows:

   a. In the domain tree, select **Services**, **Messaging**, **JMS Modules**.

   b. Click **ORM JMSModule**.

   c. Select **ORM NotificationTopic**, **ORM FinisherQueue**, **ORM LoaderQueue**, **ORM IncomingEventQueue, Finalization ORM ConnectionFactory.**

   d. Click **Delete**.

2. Recreate the Oracle Role Manager NotificationTopic as follows:

   a. Click **New**.

   b. Select **Distributed Topic**.

   c. Click **Next**.

   d. In the **Name** field, type `ORM NotificationTopic`.

   e. In the **JNDI Name** field, type `orm/topic/NotificationTopic`.

   f. Click **Finish**.

3. Recreate the Oracle Role Manager LoaderQueue as follows:

   a. Click **New**.

   b. Select **Distributed Queue**.

   c. Click **Next**.

   d. In the **Name** field, type `ORM LoaderQueue`.

   e. In the **JNDI Name** field, type `orm/queue/LoaderQueue`.

   f. Click **Finish**.

4. Recreate the Oracle Role Manager IncomingEventQueue as follows:

   a. Click **New**.

   b. Select **Distributed Queue**.

   c. Click **Next**.

    d. In the **Name** field, type `ORM IncomingEventQueue`.

    e. In the **JNDI Name** field, type `orm/queue/IncomingEventQueue`.

    f. Click **Finish**.

4.3.2.6 Configuring Finalization Server

1. Create a server, `ORMFinalizationServer` as follows:

> **Note:** The new server can be in a different domain, or in the same domain, but not in the cluster.

    a. In the domain tree, select **Environment**, **Servers**.

    b. Click **New**.

    c. In the **Server Name** field, type `ORMFinalizationServer`.

    d. In the **Server Listen Port** field, type `7074`.

    e. Click **Finish**.

2. Configure a machine for the Oracle Role Manager FinalizationServer as follows:

    a. In the domain tree, select **Environment**, **Servers**.

    b. Click **ORMFinalizationServer**.

    c. Click **Machine** and select **ORM_Machine** on which this server is to be run.

    d. Click **Save**.

3. Create a JMS Server as follows:

    a. In the domain tree, select **Services**, **Messaging, JMS Servers**.

    b. Click **New**.

    c. In the **Name** field, type `FinalizationJMSServer`.

    d. Click **Next**.

    e. In the **Target** field, select **ORMFinalizationServer**.

    f. Click **Finish**.

4. Create a JMS module as follows:

    a. In the domain tree, select **Services**, **Messaging**, **JMS Modules**.

    b. Click **New**.

    c. In the **Name** field, type `FinalizationJMSModule`.

    d. Click **Next**.

    e. In the **Target** field, select **ORMFinalizationServer**.

    f. Click **Next**.

    g. Click **Finish**.

5. Create a Subdeployment in the JMS module as follows:

    a. In the domain tree, select **Services**, **Messaging**, **JMS Modules**.

    b. Click **FinalizationJMSModule**.

    c. Click the **Subdeployments** tab.

    d. Click **New**.

    e. In the **Subdeployment Name** field, type `ORM JMSSubdeployment`.

    f. Click **Next** and select **FinalizationJMSServer** as the target JMS server.

    g. Click **Finish**.

6. Create a connection factory as follows:

    a. In the domain tree, select **Services**, **Messaging**, **JMS Modules**.

    b. Click **FinalizationJMSModule**.

    c. Click **New**.

    d. Select **Connection Factory** and click **Next**.

    e. In the **Name** field, type `Finalization ORM ConnectionFactory`.

    f. In the **JNDI Name** field, type `orm/remote/jms/FinalizationConFac`.

    g. Click **Next** and then click **Finish**.

    h. Click **Finalization ORM ConnectionFactory**.

    i. Click the **Transactions** tab.

    j. Select **XA Connection Factory Enabled**.

    k. Click **Save**.

7. Create a queue as follows:

    a. In the domain tree, select **Services**, **Messaging**, **JMS Modules**.

    b. Click **FinalizationJMSModule**.

    c. Click **New**.

    d. Select **Queue** and click **Next**.

    e. In the **Name** field, type `ORM FinalizationQueue`.

    f. In the **JNDI Name** field, type `orm/remote/queue/BtFinisherQueue`.

    g. Click **Next**.

    h. In the **Subdeployments** field, select `ORM JMSSubdeployment`.

    i. Click **Finish**.

8. Change the configuration of Oracle Role Manager Data Source to target ORMFinalizationServer as follows:

    a. In the domain tree, select **Services, JDBC, Data Sources**.

    b. Click **ORM Data Source**.

    c. Click the **Targets** tab.

    d. Select **ORMFinalizationServer**, and ensure that **ORMCluster** is selected.

    e. Click **Save**.

9. Create a JDBC Store as follows:

    a. In the domain tree, select **Services**, **Persistent Stores**.

    b. Click **New** and select **Create JDBCStore** from the list.

    c. In the **Name** field, type `ORMJDBCStoreF`.

    d. In the **Target** field, select **ORMFinalizationServer**.

    e. In the **DataSource** field, select **ORM Data Source**.

    f. In the **Prefix Name** field, type `ORMF`.

    g. Click **OK**.

10. Create a foreign server as follows:

    a. In the domain tree, select **Services**, **Messaging**, **JMS Modules**.

    b. Click **ORM JMSModule** and click **New**.

    c. Select **Foreign Server** and click **Next**.

    d. In the **Name** field, type `FinalizationServer`, click **Next** and then click **Finish**.

    e. Click **FinalizationServer**.

    f. In the **JNDI Initial Context Factory** field, type `weblogic.jndi.WLInitialContextFactory`.

    g. In the **JNDI Connection URL** field, type `t3://machine_name:7074.`

> **Note:** The machine_name is the name of the machine where ORMFinalizationServer is deployed. You must not use localhost as a machine name.

    h. Select **Default Targeting Enabled** and click **Save**.

    i. Click the **Destinations** tab and click **New**.

    j. In the **Name** field, type `ORM FinalizationQueue`.

    k. In the **Local JNDI Name** field, type `orm/queue/BtFinisherQueue`.

    l. In the **Remote JNDI Name** field, type `orm/remote/queue/BtFinisherQueue` and click **OK**.

    m. Click the **Connection Factories** tab and click **New**.

    n. In the **Name** field, type `ORM Finalization ConnectionFactory`.

    o. In the **Local JNDI Name** field, type `orm/jms/FinalizationConFac`.

    p. In the **Remote JNDI Name** field, type `orm/remote/jms/FinalizationConFac` and click **OK**.

4.3.2.7 Configuring Connection Factory

Configure the connection factory as follows:

1. In the domain tree, select **Services**, **Messaging**, **JMS Modules**.

2. Click **ORM JMSModule**.

3. Click **ORM ConnectionFactory**.

4. Deselect **Default Targeting Enabled**.

5. Click **Save**.

6. Click **Subdeployment** tab.

7. In the **Subdeployment** field, select **cf-sub** and click **Save**.

4.3.2.8 Setting the Target

Change the deployed applications (ORMServerApp and webui) to ORMCluster `All servers in the cluster` as target.

4.3.2.9 Ensuring the Default Target Enabled for Topics and Queues

Ensure that ORM IncomingEventQueue, ORM LoaderQueue, and ORM NotificationTopic have Default Targeting enabled by performing the following steps:

1. In the domain tree, select **Services**, **Messaging**, **JMS Modules**.

2. Click **ORM JMSModule**.

3. Click on **ORM IncomingEventQueue**.

4. Ensure that the Default Targeting Enabled is selected.

5. Click **Save**.

6. Repeat the same for ORM LoaderQueue and ORM Notification Topic.

4.3.2.10 Configuring SSL

Configure SSL for the admin server as follows:

1. In the domain tree, select **Environment**, **Servers**.

2. Click **AdminServer(admin)**.

3. Click the **SSL** tab.

4. Click **Advanced**.

5. In the **Hostname Verification** field, select **None**.

6. Click **Save**.

> **Note:** You must shutdown all servers after performing the preceding step

4.3.2.11 Pack/unpack the Domain to Secondary Node

> **Note:** Ensure that all server node managers are running while performing the following steps.

1. On primary node, run the following command:

```
>cd WLS_HOME/common/bin
>pack.cmd -domain=$DOMAIN_HOME -template=/tmp/template_x.jar
-template_name="template_x" -managed=true
```

2. Copy template_x.jar to secondary node.

   On secondary node, run the following command:

```
>cd $WLS_HOME/common/bin
>unpack.cmd -template=??/template_x.jar -domain=$DOMAIN_HOME
```

> **Note:** Start ORMFinalizationServer before starting the Managed Server.

4.3.3 Configuring Apache Failover Proxy

To configure Apache failover proxy:

1. Install Apache HTTP server 2.2.

2. Download the WebLogic Apache plug-in from:

   http://www.oracle.com/technology/products/weblogic/index.html

3. Unzip and copy:

   win/mod_wl_22.so to apache_home/modules directory

4. Add the following to httpd.conf file by substituting the values for IP addresses and ports for your environment:

   ```
   LoadModule weblogic_module modules/mod_wl_22.so

   <IfModule mod_weblogic.c>
     WebLogicCluster node1_ip:port,node2_ip:port
   </IfModule>

   <Location /webui>
     SetHandler weblogic-handler
   </Location>

   <Location /ormconsole>
     SetHandler weblogic-handler
   </Location>
   ```

5. Restart Apache and all WebLogic servers.

You must be able to access webui and console at:

http://myApacheServer/webui

http://myApacheServer/ormconsole

For more information about installing and configuring the Apache HTTP Server Plug-In, refer to the following URL:

http://e-docs.bea.com/wls/docs103/plugins/apache.html

5

Configuring IBM WebSphere

This chapter contains procedures for configuring the IBM WebSphere application servers for Oracle Role Manager in preparation for deployment of the Oracle Role Manager on either nonclustered or clustered server environments. The procedures in this chapter are expected to be performed in the sequence they are presented.

This chapter includes the following sections:

- Preparing WebSphere for a Nonclustered Server Installation
- Preparing WebSphere for a Clustered Server Installation
- Configuring JDBC Providers and Data Sources
- Configuring JMS Messaging Buses and Bus Destinations
- Configuring JMS Queues and Connection Factories
- Configuring JMS Activation Specifications
- Configuring Security
- Configuring Data Upload Size Limit
- Increasing the Transaction Timeout
- Setting Up the Server Virtual Host Information (Clustered Environment Only)
- Deploying Oracle Role Manager

5.1 Preparing WebSphere for a Nonclustered Server Installation

This procedure assumes that a WebSphere application server profile has been created for Oracle Role Manager with a host alias set for port access to Oracle Role Manager.

> **Note:**
>
> - During profile creation, you must select the option to enable administrative security.
> - When configuring WebSphere, it is recommended that you save your settings after every task.

You must configure IBM WebSphere server in SSL mode to operate in a secure environment. For information about configuring SSL for WebSphere server, refer to the following URL:

```
http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?
topic=/com.ibm.websphere.ihs.doc/info/ihs/ihs/welc6topsecureihs.
html
```

This section includes the following topics:

- Creating a Non-Administrative Server for Deploying Oracle Role Manager
- Configuring WebSphere to Use a Non-Default HTTP Port

5.1.1 Creating a Non-Administrative Server for Deploying Oracle Role Manager

Oracle recommends you to deploy the Oracle Role Manager system on a non-administrative server.

To create a non-administrative server:

1. Run the following command:

    ```
    [WebSphere Install Dir]/AppServer/profiles/[Profile name]/bin/wsadmin.bat
    ```

    > **Note:** You must ensure that the WebSphere server is running before performing this step.

2. On the wsadmin prompt (wsadmin>), enter the following commands:

    ```
    $AdminTask createApplicationServer <Websphere Node Name> { -name orm
    -templateName default }
    $AdminConfig save
    quit
    ```

    > **Note:** The node name specified in the first command must be same as the node name of the administrative server that gets created by default, for example server1. You can find out the node name on the admin console by going to Servers, Application Servers.

5.1.2 Configuring WebSphere to Use a Non-Default HTTP Port

If you are deploying the system on a non-administrative server, then perform the following steps:

To configure the WebSphere application server to use a non-default port:

1. If not already on the WebSphere administrative console, in a Web browser, type the URL, for example:

    ```
    http://<appserverhost>:9060/ibm/console
    ```

2. Select **Environment**, **Virtual Host**, **default_host**, **Host Aliases** and then click **New**.

3. In the **Host Name** field, type *.

4. In the **Port** field, enter the HTTP port number of the non-admin server on which Oracle Role Manager is going to be deployed, for example 9081.

5. Click **OK**.

5.2 Preparing WebSphere for a Clustered Server Installation

This section describes the steps to prepare WebSphere for deployment of Oracle Role Manager in a clustered application server environment.

> **Caution:** Deploying an application in a clustered environment is a complex procedure. This document assumes that you have expertise in installing and using applications in a WebSphere cluster. These instructions provide the Oracle Role Manager-specific details only. They are not complete instructions for setting up a WebSphere cluster. For more information about clustering, refer to WebSphere documentation.

This section includes the following topics:

- Installing Network Deployment Manager
- Upgrading NDM
- Preparing the Database
- Creating Profiles on Application Servers
- Setting Up the Servers
- Creating the Cluster

5.2.1 Installing Network Deployment Manager

You need to install Network Deployment Manager (NDM) if you are setting up the WebSphere cluster. Install NDM on the same computer where WebSphere Application Server is installed.

To install NDM:

1. Launch the NDM installer (double click Install.exe).

2. Specify the following information:

   a. Select the product installation directory.

   b. Under WebSphere Application Server environments, choose the **None** option.

3. Continue with the installation. When the NDM installer launches the WebSphere "First Steps" application, exit it and finish the installation.

5.2.2 Upgrading NDM

To upgrade the NDM from 6.1 to 6.1.0.21:

1. Install the update installer to fix pack 21.

2. Accept default values.

5.2.3 Preparing the Database

Preparing the database includes:

- Creating the database for finalization bus
- Creating the database for each planned server

To create the database user for finalization bus:

To create the database user for finalization bus, for example, WSMsgFin, use the following SQL commands:

```
create user WSMsgFin identified by <password>
default tablespace ORM_DATA
temporary tablespace ORM_TEMP;
grant connect to WSMsgFin;
grant create session to WSMsgFin;
grant resource to WSMsgFin;
commit;
```

To create the database user for each planned server:

To create the database user for each planned server in the cluster, for example, WSMsgEng1 and WSMsgEng2, run the following SQL commands:

```
create user WSMsgEng1 identified by <password>
default tablespace ORM_DATA
temporary tablespace ORM_TEMP;
grant connect to WSMsgEng1;
grant create session to WSMsgEng1;
grant resource to WSMsgEng1;
create user WSMsgEng2 identified by <password>
default tablespace ORM_DATA
temporary tablespace ORM_TEMP;
grant connect to WSMsgEng2;
grant create session to WSMsgEng2;
grant resource to WSMsgEng2;
commit;
```

5.2.4 Creating Profiles on Application Servers

To create primary profile on server1:

1. Select **Start**, **Programs**, **IBM WebSphere**, **Application Server Network Deployment**, and then select **Profile Management tool**.

2. Select **Cell (deployment manager and a federated application server)**, and then click **Next**.

3. Select **Typical Profile Creation**, and then click **Next**.

4. Select **Enable administrative security**, enter the user name and password and click **Next**.

5. Click **Create** to create the primary profile.

To create secondary profile on server2:

1. Select **Start**, **Programs**, **IBM WebSphere**, **Application Server Network Deployment**, and then select **Profile Management tool**.

2. Select **Custom Profile** for Environments and click **Next**.

3. Select **Advanced Profile creation**, and then click **Next**.

4. In the **Profile Name** field, enter the profile name and click **Next**.

5. In the **Node Name** field, enter the Hostname of the computer and click **Next**.

6. Select **Federate this node later**, and then click **Next**.

7. Click **Create** to create the secondary profile.

5.2.5 Setting Up the Servers

To start the deployment manager:

1. On server1, in the command prompt, change to the
    WEBSPHERE_HOME\profiles*DEPLOYMENT_MANAGER_PROFILE_NAME*\bi
    n directory

    Where:

    - *WEBSPHERE_HOME* is the home directory of WebSphere

    - DEPLOYMENT_MANAGER_PROFILE_NAME is the name of the
      deployment manager profile being used

    For example:

    ```
    C:\Program Files\IBM\WebSphere\AppServer\profiles\Dmgr01\bin
    ```

2. Run the following command:

    ```
    startManager.bat
    ```

    Note: Before performing the following procedure, ensure that the
    WebSphere application server is down.

To start the node:

1. On server1, in the command prompt, change to the
    WEBSPHERE_HOME\profiles*PRIMARY_PROFILE_NAME*\bin directory

    Where:

    - *WEBSPHERE_HOME* is the home directory of WebSphere

    - *PRIMARY_PROFILE_NAME* is the name of the primary application server
      profile being used

    For example:

    ```
    C:\Program Files\IBM\WebSphere\AppServer\profiles\AppSrv01\bin
    ```

2. Run the following command:

    ```
    startNode.bat
    ```

To add a node:

1. On server2, in the command prompt, change to the
    WEBSPHERE_HOME\profiles*SECONDARY_PROFILE_NAME*\bin directory

    Where:

    - *WEBSPHERE_HOME* is the home directory of WebSphere

    - *SECONDARY_PROFILE_NAME* is the name of the secondary application
      server profile being used

    For example:

    ```
    C:\Program Files\IBM\WebSphere\AppServer\profiles\Custom01\bin
    ```

2. Run the following command:

```
addNode.bat <Primary_Node_Machine_Name>
<Cell_Deployment_Manager_SOAP_Connector_Port> -username websphere -password
websphere
```

5.2.6 Creating the Cluster

To create the cluster:

1. Using a Web browser, connect to the Network Deployment Manager administrative console by navigating to the following URL:

   http://NDM_HOST:NDM_PORT/admin

2. Log on to the system.

3. Click **Servers** in the left panel.

4. Click **Clusters**.

5. Click **New**.

   a. Enter the cluster name, for example, ORM Cluster.

   b. Ensure that you select the **Prefer local** and **Configure HttpSession memory-to-memory replication** check boxes, and then click **Next**.

6. Enter the first node member's name, for example, ORM Server1 and click **Next**.

7. Click **Add Member**.

8. Enter a name for the second node member, for example ORM Server2.

9. Select the node for the second node member, then click **Add Member**.

10. Click **Next**, then click **Finish**.

5.2.7 Distributing the Oracle Role Manager Libraries

For clustered server environments, the Oracle Role Manager libraries must exist in the identical location on all nodes where the Oracle Role Manager server exists.

To distribute the Oracle Role Manager libraries:

1. On the primary server where Oracle Role Manager is installed, navigate to the *ORM_HOME*/lib directory.

2. Make a note of the full path, for example, C:\oracle\orm\lib.

3. On each node for Oracle Role Manager, create a directory and path that exactly matches the path in the previous step.

4. Copy all of the files from *ORM_HOME*/lib into each of the directories created in the previous step.

5.3 Configuring JDBC Providers and Data Sources

This section includes the following topics:

- Configuring JDBC Providers
- Reconfiguring JDBC Providers
- Creating the Oracle Role Manager Database Credentials

- Configuring the Non-XA Data Source

- Configuring the Transaction (XA) Data Source

- Configuring the Messaging Engine (XA) Data Source (Clustered Environments Only)

5.3.1 Configuring JDBC Providers

To configure the transaction (XA) and non-transaction JDBC providers:

1. If the Oracle Role Manager database is Oracle 11g, copy the JDBC driver as follows:

   a. On the Oracle database host, navigate to *ORA_HOME*/jdbc/lib.

   b. Copy the ojdbc5.jar file from *ORACLE_HOME*/jdbc/lib into *ORM_HOME*/lib on the application server host.

   c. For clustered server environments, repeat these steps so the JDBC driver exists in *ORM_HOME*/lib on all nodes where Oracle Role Manager servers exist.

2. If not already on the WebSphere administrative console, in a Web browser, type the URL, for example:

   `http://<appserverhost>:9060/ibm/console`

3. From **Resources**, select **JDBC**, then click **JDBC Providers**.

4. For nonclustered configuration, select the cell scope, Node=*node_name*, Server=*server_name*, from the **Scope** list.

5. For clustered configuration, select the cluster scope, Cluster=*cluster_name*, from the **Scope** list.

6. Click **New** to create the XA JDBC provider.

7. Select **Oracle** as the database type.

8. Select **Oracle JDBC Driver** as the provider type.

9. Select **XA datasource** as the Implementation type, and then click **Next**.

10. In the **Directory location** field, type the full path to the directory containing the Oracle JDBC driver.

    For Oracle 10g, the correct driver is ojdbc14.jar. This driver is contained in *ORM_HOME*/lib, so enter that path.

    For Oracle 11g, the correct driver is ojdbc5.jar. This driver was copied to *ORM_HOME*/lib Step 1, so enter that path.

11. Click **Next**, then **Finish**.

12. Click **New** to create the non-XA JDBC provider.

13. Select **Oracle** as the database type.

14. Select **Oracle JDBC Driver** as the provider type

15. Select **Connection pool data source** as the Implementation type, and then click **Next**.

16. Ensure that the value in the **Directory location** field is correct, and then click **Next**.

17. Click **Finish**.

5.3.2 Reconfiguring JDBC Providers

This section is applicable only when you are using Oracle 11g JDBC driver (ojdbc5.jar) to connect to oracle 11g database.

Oracle recommends you to use ojdbc5.jar when configuring Oracle Role Manager with Oracle11g database. WebSphere Application Server by default creates Oracle JDBC providers using the Oracle 10g JDBC driver (ojdbc14.jar). In the administrative console, the wizard for creating new data sources does not allow you to change the name of the jar file. For example, you cannot change the entry from ojdbc14.jar to ojdbc5.jar. For more information about JDBC providers, refer to the following link:

http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?
topic=/com.ibm.websphere.base.doc/info/aes/ae/rdat_minreqoracle.
html

After you create the JDBC provider using the wizard, modify it to change the class path entry to reflect the location of ojdbc5.jar as follows:

To reconfigure the JDBC providers:

This procedure assumes that you have already performed the steps mentioned in Section 5.3.1.

1. If not already on the WebSphere administrative console, in a Web browser, enter the URL, for example: http://<appserverhost>:9060/ibm/console.

2. In the administrative console, go to Resources, JDBC, JDBC Providers.

3. Select the cell scope from the **Scope** list, and then click on JDBC provider, **Oracle JDBC Driver**.

4. In the Class path field, enter the full path of the JDBC drivers ojdbc5.jar, for example, *ORM_HOME*/lib/ojdbc5.jar.

5. Click **Apply**.

6. Click **Save**.

7. Repeat the steps from 2 to 6 for the JDBC provider, Oracle JDBC Driver (XA).

> **Note:** You must execute these steps before creating any data sources as mentioned in "Configuring the Non-XA Data Source" on page 5-9 and "Configuring the Transaction (XA) Data Source" on page 5-10. If the data sources are already created, then you must recreate after deleting them.

5.3.3 Creating the Oracle Role Manager Database Credentials

To create the Oracle Role Manager database alias:

1. From **Security**, select **Secure administration, applications, and infrastructure**.

2. In the Authentication area, select **Java Authentication and Authorization Service** and then click the **J2C authentication data** link.

3. Click **New**.

4. Enter a name for the alias, for example, ORM Database to identify the Oracle Role Manager database.

5. Type the user ID and password for the Oracle Role Manager application user as specified in Section 3.2.2.

6. Click **OK**.

7. For clustered environments, repeat this procedure for other messaging engines in the cluster, for example, WSMsgFin, WSMsgEng1, and WSMsgEng2.

> **Note:** The messaging engine names and passwords must match those that were used in Section 5.2.3 when preparing the database.

5.3.4 Configuring the Non-XA Data Source

To configure the non-XA data source and credentials:

1. Go to Resources, JDBC, Data sources.

2. For nonclustered server environments, select the same cell scope used in Section 5.3.1, then click **New**.

3. For clustered server environments, select the same cluster scope used in Section 5.3.1, then click **New**.

4. In the **Data source name** field, type a name for the non-XA data source, for example, `ORM Non-XA Data source`.

5. In the **JNDI name** field, type `orm/jdbc/ORMServerDS`, and then click **Next**.

6. Choose an existing JDBC provider and select the non-XA JDBC provider you created in Section 5.3.1, for example, **Oracle JDBC Driver**, and then click **Next**.

7. In the **URL** field, type the JDBC connection URL:

   `jdbc:oracle:thin:@<server>[:<port>]:<database_name>`

   For example:

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

8. Select **Oracle10g data store helper** from the list, and then click **Next.**

> **Note:** You must use Oracle11g data store helper for Oracle 11g database.

9. Click **Finish**.

   The non-XA data source for Oracle Role Manager should appear in the list.

10. Click the name of the new non-XA data source to display details.

11. In the Container-managed authentication alias list, select the database alias created in Step 4 of Section 5.3.3 and then click **Apply**.

   The reference to this option being deprecated can be ignored.

12. On the same page, in the Additional Properties section, click Connection Pool Properties and set the Maximum connections to `30`.

13. Click **OK**.

   The reference to the failed test connection can be ignored because the connection will work after restarting the deployment manager.

5.3.5 Configuring the Transaction (XA) Data Source

To configure the XA data source:

1. Go to Resources, JDBC, Data sources.

2. For nonclustered server environments, select the same cell scope used in Section 5.3.1, then click **New**.

3. For clustered server environments, select the same cluster scope used in Section 5.3.1, then click **New**.

4. In the **Data source name** field, type a name for the XA data source, for example, `ORM XA Data source`.

5. In the **JNDI name** field, type `orm/jdbc/ORMServerXADS`, and then click **Next**.

6. Choose an existing JDBC provider and select the XA JDBC provider that you created in Section 5.3.1, for example, **Oracle JDBC Driver (XA)**, and then click **Next**.

7. In the **URL** field, type the JDBC connection URL:

    `jdbc:oracle:thin:@<server>[:<port>]:<database_name>`

    For example:

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

8. Select **Oracle10g data store helper** from the list, and then click **Next**.

    > **Note:** You must use Oracle11g data store helper for Oracle 11g database.

9. Click **Finish**.

    Both the new XA data source and non-XA data source for Oracle Role Manager must appear in the list.

10. Click the name of the XA data source to display details.

11. In the Container-managed authentication alias list, select the database alias created in Section 5.3.3, and then click **Apply**.

    The reference to this option being deprecated can be ignored.

12. On the same page, in the Additional Properties section, click **Connection Pool Properties** and set the Maximum connections to `30`.

13. Click **OK**.

    The reference to the failed test connection can be ignored because the connection will work after restarting the deployment manager.

5.3.6 Configuring the Messaging Engine (XA) Data Source (Clustered Environments Only)

> **Note:** This is section is only for clustered configuration, perform the steps mentioned in this section following the order listed above.

To configure the messaging engine (XA) data source:

1. Go to Resources, JDBC, Data sources.

2. Select the same cluster scope used in Section 5.3.1, then click **New**.

3. In the **Data source name** field, type a name for the XA data source, for example, `ORM WSMsgEng Data source`.

4. In the **JNDI name** field, type `orm/jdbc/WSMsgEngDS`, and then click **Next**.

5. Choose an existing JDBC provider and select the XA JDBC provider that you created in Section 5.3.1, for example, **Oracle JDBC Driver (XA)**, and then click **Next**.

6. In the **URL** field, type the JDBC connection URL:

   `jdbc:oracle:thin:@<server>[:<port>]:<database_name>`

   For example:

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

7. Select **Oracle10g data store helper** from the list, and then click **Next**.

   > **Note:** You must use Oracle11g data store helper for Oracle 11g database.

8. Click **Finish**.

   The Messaging Engine (XA) data source for Oracle Role Manager should appear in the list.

9. Click the name of the Messaging Engine (XA) data source to display details.

10. In the Additional Properties section, click **Connection Pool Properties** and set the Maximum connections to `30`.

11. Click **OK**.

    The reference to the failed test connection can be ignored because the connection will work after the authentication alias is configured.

5.4 Configuring JMS Messaging Buses and Bus Destinations

This section includes the following topics:

- Configuring the JMS messaging buses (Nonclustered Environments Only)
- Configuring the JMS messaging buses (Clustered Environments Only)
- Configuring Bus Destinations

5.4.1 Configuring the JMS messaging buses (Nonclustered Environments Only)

To configure the JMS messaging buses:

1. Select **Service integration, Buses**, and then click **New**.

2. Type a name for the Oracle Role Manager bus, such as `ORM Bus`, and deselect the **Bus security** check box, and then click **Next**.

3. Click **Finish**.

4. Click **New** to create the finalization bus.

5. Type a name for the finalization bus, such as `ORM Finalization Bus`, and deselect the **Bus security** check box, and then click **Next**.

> **Note:** ▪ If you do not set the name to "ORM Finalization Bus", you must provide the "oracle.iam.rm.finalization.WebSphereFinalizationBusName" system property with the name that you use.
>
> ▪ If you want to use a different name for the finalization bus, you must follow the Step 6 to set the WebSphereFinalizationBusName property. Otherwise, skip the next step.

---

6. If you use a name other than ORM Finalization Bus:

   a. Select **Servers, Application Servers**.

   b. Click the server on which Oracle Role Manager is installed.

   c. In the Server Infrastructure section, click **Java and Process Management, Process Definition**.

   d. Click **Java Virtual Machine**.

   e. Click **Custom Properties**.

   f. Click **New**.

   g. In the **Name** field, type the following text:

      `oracle.iam.rm.finalization.WebSphereFinalizationBusName`

   h. In the **Value** field, type the name you set for the finalization bus.

   i. Click **OK**.

   j. Click **Finish**.

7. Add the server to each of the newly created buses as follows:

   a. Click **ORM Bus** link, and then click **Bus members**.

   b. Click **Add**.

   c. Select the server to use for Oracle Role Manager, and then click **Next**.

   d. In the **Select the type of message store** list, select **File Store**, click **Next**, and then click **Next** again.

   e. Click **Finish**.

5.4.2 Configuring the JMS messaging buses (Clustered Environments Only)

To configure the JMS messaging buses:

1. Select **Service integration, Buses**, and then click **New**.

2. Type a name for the Oracle Role Manager bus, such as `ORM Bus`, and deselect the **Bus security** check box, and then click **Next**.

3. Click **Finish**.

4. Click **New** to create ORM Notification Bus, deselect the **Bus Security** check box and click **Next**.

5. Click **Finish**.

6. Click **New** to create the finalization bus.

7. Type a name for the finalization bus, such as `ORM Finalization Bus`, deselect the **Bus security** check box, and click **Next**.

Note: ▪ If you do not set the name to "ORM Finalization Bus", you must provide the "oracle.iam.rm.finalization.WebSphereFinalizationBusName" system property with the name that you use.

▪ If you want to use a different name for the finalization bus, you must follow the Step 6 to set the WebSphereFinalizationBusName property. Otherwise, skip the next step.

---

8. If you use a name other than ORM Finalization Bus:

   a. Select **Servers, Application Servers**.

   b. Click the server on which Oracle Role Manager is installed.

   c. In the Server Infrastructure section, click **Java and Process Management, Process Definition**.

   d. Click **Java Virtual Machine**.

   e. Click **Custom Properties**.

   f. Click **New**.

   g. In the **Name** field, type the following text:

   `oracle.iam.rm.finalization.WebSphereFinalizationBusName`

   h. In the **Value** field, type the name you set for the finalization bus.

   i. Click **OK**.

   j. Click **Finish**.

9. Add the server to the ORM Bus as follows:

   a. Click the **ORM Bus** link, then click **Bus members**.

   b. Click **Add**.

   c. Select the **ORM Cluster** (not either of servers) to use for Oracle Role Manager and click **Next**.

   d. In the **Select the type of message store** list, select **Data Store**, then click **Next**.

   e. Specify the following values:

   JNDI name: `orm/jdbc/WSMsgEngDS`

   Schema name: `WSMsgEng1`

   Authentication alias: `WSMsgEng1`

   f. Click **Next**, then click **Finish**.

   g. Go to ORM Bus, Bus Members, and ORM Cluster.

   h. Click **Add Messaging Engine**.

   i. Select **Data Store**, then click **Next**.

    j. Specify the following values:

        JNDI name: `orm/jdbc/WSMsgEngDS`

        Schema name: `WSMsgEng2`

        Authentication alias: `WSMsgEng2`

    k. Click **Next**, then click **Finish**.

10. Add the server to the ORM Finalization Bus as follows:

    a. Click the **ORM Finalization Bus** link, then click **Bus members**.

    b. Click **Add**.

    c. Select the **ORM Cluster** (not either of servers) to use for Oracle Role Manager, then click **Next**.

    d. In the **Select the type of message store** list, select **Data Store**, then click **Next**.

    e. Choose the **Use existing data source** option and specify the following values:

        JNDI name: `orm/jdbc/WSMsgEngDS`

        Schema name: `WSMsgFin`

        Authentication alias: `WSMsgFin`

    f. Click **Next**, then click **Finish**.

11. Add the server to the ORM Notification Bus as follows:

    a. Click the **ORM Notification Bus** link, and then click **Bus members**.

    b. Click **Add**.

    c. Select the **ORM Cluster** (not either of servers) to use for Oracle Role Manager and click **Next**.

    d. In the **Select the type of message store** list, select **File Store**, then click **Next**.

    e. Specify the log directory path, for example, `C:\WSlogs`.

    f. Specify the permanent directory path, for example, `C:\WSlogs`.

    g. Click **Next**, then click **Finish**.

12. Go to Servers, Core groups, Core group settings.

13. Select **DefaultCoreGroup**.

14. Click **Policies**.

15. For each server in the cluster:

    a. Click **New**.

    b. Select **One of N policy** and click **Next**.

    c. Specify the name, for example, `Server1 SIB Policy`, `Server2 SIB Policy`.

    d. Specify `120` for the Is alive timer, select the **Failback** checkbox and click **Apply**.

    e. Click **Matching Criteria** and add the following criterias:

    ■ In the **Name** field enter `IBM_hc` and in the **Value** field enter `ORM Cluster`.

    ■ In the **Name** field enter `WSAF_SIB_BUS` and in the **Value** field enter `ORM Bus`.

- In the **Name** field enter `WSAF_SIB_MESSAGING_ENGINE` and in the **Value** field enter the name of the messaging engine for the server in question. For example:

  For `Server1 SIB Policy`, in the **Value** field, enter `ORM Cluster.000-ORM Bus`.

  For `Server2 SIB Policy`, in the **Value** field, enter `Cluster.001-ORM Bus`.

- In the **Name** field enter `type` and in the **Value** field as `WSAF_SIB`.

16. Click **Preferred Servers** and specify `ORM Server1` for the Server1 SIB Policy and `ORM Server2` for the Server2 SIB Policy.

17. Click **OK**.

5.4.3 Configuring Bus Destinations

To configure the Oracle Role Manager bus and finalization bus destinations:

1. If not already on the Buses page, go to Service integration, Buses.

2. Click the **ORM Bus** link, and then click **Destinations**.

3. Click **New**.

4. Select **Queue** as the destination type, and then click **Next**.

5. Type `Loader Queue` as the identifier, and then click **Next**.

6. Specify the bus member to own the queue, and then click **Next**.

7. Click **Finish**.

8. Repeat these steps, but this time, type `Incoming Event Queue` as the identifier.

9. For nonclustered server environments:

   a. Click **New**.

   b. Select **Topic space**, then click **Next**.

10. For clustered server environments:

    a. Click **ORM Notification Bus**.

    b. Click **Destinations**, then click **New**.

    c. Select **Topic space**, then click **Next**.

11. Type `Notification Topic` as the identifier, and then click **Next**.

12. Click **Finish**.

13. Click **ORM Finalization Bus,** then click **Destinations**

    > **Note:** The bus name you select here must match the finalization bus created in , for nonclustered environments or , for clustered environments.

14. Click **New**.

15. Choose **Queue** as the destination type, and then click **Next**.

16. Type `Finisher Queue` as the identifier, and then click **Next**.

17. Specify the bus member to own the queue, and then click **Next**.

18. Click **Finish**.

5.5 Configuring JMS Queues and Connection Factories

This section includes the following topics:

- Configuring JMS Queue Connection Factories
- Configuring the JMS Topic Connection Factory
- Configuring JMS Queues
- Configuring the JMS Notification Topic

5.5.1 Configuring JMS Queue Connection Factories

To configure JMS queue connection factories:

1. Go to Resources, JMS, Queue connection factories.

2. Select the same cell or cluster scope used in Section 5.3.1, then click **New**.

3. Choose **Default messaging provider**, and then click **OK**.

4. In the **Name** field, type a name for the Oracle Role Manager connection factory, such as `ORM QCF`.

5. In the **JNDI name** field, type `orm/jms/QueueConFac`.

6. In the **Bus name** list, select **ORM Bus**, and then click **OK**.

7. Click **New**.

8. Select **Default messaging provider**, and then click **OK**.

9. In the **Name** field, type a name for the Oracle Role Manager connection factory for finalization, such as `ORM Finalization QCF`.

10. In the **JNDI name** field, type `orm/jms/FinalizationQueueConFac`.

11. In the **Bus name** list, select **ORM Finalization Bus**.

> **Note:** The bus name you select here must match the finalization bus created in Section 5.4.1, for nonclustered environments or Section 5.4.2, for clustered environments.

12. Click **OK**.

5.5.2 Configuring the JMS Topic Connection Factory

To configure the JMS topic connection factory:

1. Go to Resources, JMS, Topic connection factories.

2. Select the same cell or cluster scope used in Section 5.3.1, then click **New**.

3. Select **Default messaging provider**, and then click **OK**.

4. In the **Name** field, type a name for the Oracle Role Manager topic connection factory, such

as `ORM TCF`.

5. In the **JNDI name** field, type `orm/jms/TopicConFac`.

6. For nonclustered server environments, in the **Bus name** list, select **ORM Finalization Bus**, then click **OK**.

7. For clustered server environments:

   a. In the **Bus name** list, select **ORM Notification Bus**, then click **OK**.

   b. In the **Durable Subscription Home** field, select **ORM Cluster.000-ORM Notification Bus**, then click **OK**.

5.5.3 Configuring JMS Queues

To configure the Loader queue:

1. Go to Resources, JMS, Queues.

2. Select the same cell or cluster scope used in Section 5.3.1, and then click **New**.

3. Select **Default messaging provider**, then click **OK**.

4. In the **Name** field, type `ORM Loader`.

5. In the **JNDI name** field, type `orm/jms/LoaderQueue`.

6. In the **Bus name** list, select **ORM Bus**.

7. In the **Queue name** list, select **Loader Queue**.

8. Click **OK**.

To configure the Incoming Event queue:

1. Go to Resources, JMS, Queues.

2. Select the same cell or cluster scope used in Section 5.3.1, then click **New**.

3. Choose **Default messaging provider**, and then click **OK**.

4. In the **Name** field, type `ORM Incoming Event Queue`.

5. In the **JNDI name** field, type `orm/jms/IncomingEventQueue`.

6. In the **Bus name** list, select **ORM Bus**.

7. In the **Queue name** list, select **Incoming Event Queue**.

8. Click **OK**.

To configure the Finalization queue:

1. Go to Resources, JMS, Queues, select the same cell or cluster scope used in Section 5.3.1, then click **New**.

2. Select **Default messaging provider**, and then click **OK**.

3. In the **Name** field, type `ORM Finisher Queue`.

4. In the **JNDI name** field, type `orm/jms/FinisherQueue`.

5. In the **Bus name** list, select **ORM Finalization Bus.**

> **Note:** The bus name you select here must match the finalization bus created in Section 5.4.1, for nonclustered environments or Section 5.4.2, for clustered environments.

6. In the **Queue name** list, select **Finisher Queue**.

7. Click **OK**.

5.5.4 Configuring the JMS Notification Topic

To configure the Notification Topic:

1. Go to Resources, JMS, Topics.

2. Select the same cell or cluster scope used in Section 5.3.1, then click **New**.

3. Select **Default messaging provider**, and then click **OK**.

4. In the **Name** and the **Topic Name** fields, type `ORM Notification Topic`.

5. In the **JNDI name** field, type `orm/jms/NotificationTopic`.

6. For nonclustered server environments, in the **Bus name** list, select **ORM Bus**.

7. For clustered server environments, in the **Bus name** list, select **ORM Notification Bus**.

8. In the **Topic space** list, select **Notification Topic**.

9. Click **OK**.

5.6 Configuring JMS Activation Specifications

To configure the Loader AS:

1. Go to Resources, JMS, Activation specifications.

2. Select the same cell or cluster scope used in Section 5.3.1, then click **New**.

3. Select **Default messaging provider**, and then click **OK**.

4. In the **Name** field, type `ORM Loader AS`.

5. In the **JNDI name** field, type `orm/jms/LoaderAS`.

6. In the **Destination type** list, select **Queue**.

7. In the **Destination JNDI name** field, type `orm/jms/LoaderQueue`.

8. In the **Bus name** list, select **ORM Bus**.

9. Click **OK**.

To configure the Incoming Event AS:

1. Go to Resources, JMS, Activation specifications.

2. Select the same cell or cluster scope used in Section 5.3.1, then click **New**.

3. Select **Default messaging provider**, and then click **OK**.

4. In the **Name** field, type `ORM Incoming Event AS`.

5. In the **JNDI name** field, type `orm/jms/IncomingEventAS`.

6. In the **Destination type** list, select **Queue**.

7. In the D**estination JNDI name** field, type orm/jms/IncomingEventQueue.

8. In the **Bus name list**, select **ORM Bus**.

9. Click **OK**.

To configure the Finisher AS:

1. Go to Resources, JMS, Activation specifications.

2. Select the same cell or cluster scope used in Section 5.3.1, then click **New**.

3. Choose **Default messaging provider**, and then click **OK**.

4. In the **Name** field, type ORM Finisher AS.

5. In the **JNDI name** field, type orm/jms/FinisherAS.

6. In the **Destination type** list, select **Queue**.

7. In the **Destination JNDI name** field, type orm/jms/FinisherQueue.

8. In the **Bus name** list, select **ORM Finalization Bus**.

> **Note:** The bus name you select here must match the finalization bus created in Section 5.4.1, for nonclustered environments or Section 5.4.2, for clustered environments.

9. In the **Maximum concurrent endpoints** field, set the value to 1.

> **Note:** You must set the value of Maximum concurrent endpoints as 1 to ensure the ORM application to function properly.

10. Click **OK**.

To configure the Notification AS:

1. Go to Resources, JMS, Activation specifications.

2. Select the same cell or cluster scope used in Section 5.3.1, then click **New**.

3. Choose **Default messaging provider**, and then click **OK**.

4. In the **Name** field, type ORM Notification AS.

5. In the **JNDI name** field, type orm/jms/NotificationAS.

6. In the **Destination type** list, select **Topic**.

7. In the **Destination JNDI** name field, type orm/jms/NotificationTopic.

8. For nonclustered server environments, in the **Bus name** list, select **ORM Bus**.

9. For clustered server environments, in the **Bus name** list, select **ORM Notification Bus**.

10. Click **OK**.

5.7 Configuring Security

This section includes the following topics:

- Securing the WebSphere Installation/Console
- Creating a Custom User
- Creating Alias for Custom User
- Configuring Connection Factory Authentication
- Configuring Activation Specification Authentication
- Securing the Message Bus

5.7.1 Securing the WebSphere Installation/Console

To secure the WebSphere installation/console:

1. In a Web browser, type the URL to connect to the WebSphere administrative console. For example:

   ```
   http://<appserverhost>:9060/ibm/console
   ```

2. Go to Security, Secure administration, applications, and infrastructure.

3. Click **Security Configuration Wizard**.

4. Select **Enable application security** and **Use Java 2 security to restrict application access to local resources** check boxes and then click **Next**.

5. In the **Select user repository**: section, select **Federated repositories** and then click **Next**.

6. Type username and password for administrative user, for example, `websphere/websphere` and then click **Next**.

   > **Note:** The username and password specified in this step must be the same as the username and password used to log in to the administrative console.

7. Click **Finish**.

8. Clear the **Warn if applications are granted custom permissions** check box.

9. Click **Apply**, and then save your changes.

10. Configure the server for Oracle Role Manager as follows:

    a. From **Servers**, select **Application Servers**.

    b. Select the server for Oracle Role Manager.

    > **Note:** For clustered environments, the server is on the cluster to use for Oracle Role Manager.

    c. In the Server Infrastructure section, click **Java and Process Management**, then click **Process Definition**.

    d. In the Additional Properties section, click **Java Virtual Machine**.

    e. In the Additional Properties section, click **Custom Properties**.

    f. Click **New**.

g. In the **Name** field, type `com.ibm.websphere.java2secman.nolog`.

h. In the **Value** field, type `true`.

i. In the **Description** field, type `Stop over_logging of security warnings`.

j. Click **OK**, and then click **Save**.

k. For clustered environments, repeat the substeps in Step 10 for all additional servers on the cluster.

5.7.2 Creating a Custom User

To create a custom user:

1. Log in to the Administrative Console.

2. Go to Users and Groups, Manage Users.

3. Click **Create** and type the following:

a. User ID, for example, `ormserver`.

b. First Name, for example, `ORM`.

c. Last Name, for example, `Server`.

d. Password, for example, `ormserver`.

4. Click **Create**, and then click **Close**.

5. Go to Users and Groups, Administrative User Roles.

6. Click **Add** and perform the following substeps:

a. In the **User** field, type the user ID created in Step 3.

b. In the **Role(s)** field, select **Operator**.

c. Click **OK**.

5.7.3 Creating Alias for Custom User

To create alias for the custom user:

1. Go to Security, Secure administration, applications, and infrastructure.

2. In the Authentication section, expand **Java Authentication and Authorization Service** and then click **J2C authentication data**.

3. Click **New** and perform the following substeps:

a. In the **Alias** field, type the alias name, for example, `ormserver`.

b. In the **User** field, type the User ID that you created in Step 3 of Section 5.7.2.

c. In the **Password** field, type the password that you created in Step 3 of Section 5.7.2.

4. Click **OK**.

5.7.4 Configuring Connection Factory Authentication

To configure authentication for the connection factories:

1. Go to Resources, Resource Adapters, J2C connection factories.

2. Click the title of the connection factory.

3. Set the Container-managed authentication alias to the custom user alias created in Section 5.7.3, then click **OK**.

   The reference to this option being deprecated can be ignored.

4. Repeat these steps for each of the new connection factories, then save your changes.

5.7.5 Configuring Activation Specification Authentication

To configure authentication for the activation specifications:

1. Go to Resources, Resource Adapters, J2C activation specification.

2. Click the title of the new activation specification.

3. Set the Authentication alias to the user alias created in Section 5.7.3, then click **OK**.

4. Repeat these steps for each of the new activation specifications, and then save your changes.

5.7.6 Securing the Message Bus

To secure the message bus:

1. Go to Security, Bus Security.

2. In the Security column, for each Oracle Role Manager bus, click **Disabled**.

3. Select the **Enable bus security** check box, and then click **Apply**.

> **Note:** Enable bus security field is disabled if you have deselected Enable Bus Security field while configuring JMS messaging buses in the step 5 of Section 5.4.1 for nonclustered environments, or step 7 of Section 5.4.2 for clustered environments

4. For each Oracle Role Manager bus:

   a. Click **Enabled**.

   b. In the Additional Properties section, click **Users and groups in the bus connector role**.

5. Click **New** and select **User Name**.

6. In the **User Name** field, type the User ID that you created in Step 3 of Section 5.7.2, then click **OK**.

5.8 Configuring Data Upload Size Limit

You can upload a DAR file to load data of maximum size 10 MB into the system. If you try to load data larger than this maximum upload size, you get an error message.

`Optionally`, you can configure the maximum data upload size limit to a higher or lower value than the default settings.

To configure the data upload size limit:

1. From **Servers**, select **Application Servers**.

2. Click the server for Oracle Role Manager, for example, **ORM Server**.

3. In the **Server Infrastructure** section, expand **Java and Process Management**, and then click **Process Definition**.

4. In the **Additional Properties** section, click **Java Virtual Machine,** and then click **Custom Properties**.

5. Click **New** and type the following information:

   a. In the **Name** field, type `oracle.iam.rm.loader.max_upload_size`.

   b. In the **Value** field, type the maximum size (in bytes) of data upload that you want to set, for example, `10485760`.

   > **Note:** The default value of the maximum size of data upload is 10 MB (10x1024x1024 = 10485760). You can modify this value to any other limit.

   c. In the **Description** field, type the description for the maximum upload size that you set, for example, maximum size limit for the Oracle Role Manager loader.

   d. Click **OK**.

6. For clustered server environments, repeat these steps for each Oracle Role Manager server in the cluster.

5.9 Increasing the Transaction Timeout

To increase the transaction timeout:

1. From **Servers**, select **Application Servers**.

2. Click the server for Oracle Role Manager, for example, **ORM Server**.

3. In the **Container Settings** section, expand **Container Services**, and then Click **Transaction Service**.

4. In the **Total transaction lifetime timeout** field, type the new value, 1200, if you want to change the default value, which is `120`.

5. In the **Maximum transaction timeout** field, type the new value, 1200, if you want to change the default value, which is `300`.

6. Click **Apply** to save the server settings.

7. For clustered server environments, repeat the preceding steps for each Oracle Role Manager server in the cluster.

8. Restart the server or cluster, accordingly.

> **Note:** For performance tuning of the WebSphere Application Server, refer to the following link:
>
> http://www-01.ibm.com/software/webservers/appserv/was/performance.html

5.10 Setting Up the Server Virtual Host Information (Clustered Environment Only)

The application server uses the virtual host information setup on the Node Manager to properly configure the Web server plug-ins to distribute the load and deal with failover. When you add a server to the cluster, you must update the virtual host information.

To update the virtual host information:

1. Ensure that Node Manager is running.

2. Using a Web browser, connect to the Node Manager administrative console by navigating to the following URL:

   ```
   http://NDM_HOST:NDM_PORT/admin
   ```

3. Log in using the Oracle Identity Manager Administrator name and password that you specified during installation.

4. In the left pane, click **Servers**.

5. Click **Application Servers**, then click **ORM Server1**.

6. In the Communications section, click **Ports**.

   Make note of the port numbers for WC_defaulthost. You will need this port number for the new host alias created later in this procedure.

7. In the left pane, click **Environment**.

8. Click **Virtual Hosts**, then click **default_host**.

9. Click **Host Aliases**, then click **New**.

10. In the **Host Name** field, enter an asterisk (*).

11. In the **Port** field, enter the port that you noted as the port for WC_defaulthost, then click **Apply**.

12. Select **Preferences, Synchronize changes with Nodes**, then click **Apply**.

13. Click **Save**.

    Virtual host setup for the ORM Server1 server is complete.

14. Repeat the procedure for all available servers in ORM_CLUSTER, for example, ORM Server2 and so forth.

5.11 Deploying Oracle Role Manager

This section includes the following topics:

- Deploying the Oracle Role Manager Server
- Deploying the Oracle Role Manager Web Application

5.11.1 Deploying the Oracle Role Manager Server

To deploy the Oracle Role Manager server:

1. Go to **Applications, Install New Application**.

2. Choose **Remote file system**, click **Browse** to navigate to the *ORM_HOME*/lib directory, select server.ear, and then click **OK**.

3. Click **Next**.

4. On the Map modules to servers page, perform the following substeps:

   a. From the **Clusters and Servers** list, select the server or cluster on which Oracle Role Manager is to be deployed.

   b. Select both modules, **server.jar** and **ormconsole**.

   c. Click **Apply.**

   d. Click **Next**.

5. Click **Finish**.

   This could take a few moments to complete.

6. Click **Save**.

7. For clustered server environments, click **OK**.

To associate the custom user to the Oracle Role Manager server:

1. Go to Applications, Enterprise Applications.

2. Select **ORM Server**.

3. In the Detail Properties section, click **Security role to user/group mapping**.

4. Select **ORMServer**, and then click **Look up users**.

5. Search and select the **ormserver** user that you created in Step 3 of Section 5.7.2, then move it to the **Selected** list by clicking the right arrow and then click **OK**.

6. In the Security role to user/group mapping page, click **OK**.

7. In the Detail Properties section, click **User RunAs roles**.

8. Perform the following substeps:

   a. In the **User Name** field, type the User ID that you created in Step 3 of Section 5.7.2.

   b. In the **password** field, type the password that you created in Step 3 of Section 5.7.2.

   c. In the **Role(s)** field, select **ORMServer**.

9. Click **Apply**, and then click **OK**.

10. Restart the server or cluster on which Oracle Role Manager is installed.

> **Note:**
>
> - If you have created a non-admin server in the step 5.1.1 on page 2 for Oracle Role Manager, then the non-admin server does not get started automatically when the websphere admin server is started or when the websphere windows service is started. You can start the non-admin server using the following command:
>
>   ```
>   [Websphere Profile Install Dir]/bin/startServer.bat
>   <server-name>
>   ```
>
> - After starting the server on which Oracle Role Manager is installed, if you see the status of the Oracle Role Manager server application through the administrative console of the administrative server, it might show as stopped though the application has actually started. You cannot start the application deployed on the non-admin server through the administrative console of the administrative server.

11. To test the server installation, ensure that you can get to the Oracle Role Manager administrative console from a Web browser. For example:

    ```
    http://localhost:9080/ormconsole
    ```

    You should see the Home page of the Oracle Role Manager administrative console.

5.11.2 Deploying the Oracle Role Manager Web Application

To deploy the Oracle Role Manager Web application:

1. Go to **Applications, Install New Application**.

2. Select **Remote file system**, click **Browse** to navigate to the *ORM_HOME*/webui/websphere/6.1 directory, select webui.ear, click **OK** and then click **Next**.

3. On the Select installation options page, accept the defaults and then click **Next**.

4. On the Map modules to servers page, perform the following substeps:

   a. From the **Clusters and Servers** list, select the server or cluster on which Oracle Role Manager is to be deployed.

   b. Select the **webui** module.

   c. Click **Apply.**

   d. Click **Next**.

5. Click **Finish**, and then save your changes.

6. Go to Applications, Enterprise Applications, ORM Web UI.

7. Click **Manage Modules**.

8. Click the **webui** link.

9. In the **Class loader order** list, select **Classes loaded with application class loader first and apply**.

10. From Applications, Enterprise Applications, select **ORM Web UI**, and then click **Start**.

(This assumes you are administering WebSphere on the same server as the ORM Web UI is installed).

> **Note:** If Web UI is deployed on a non-administrative server or cluster, then restart that server or cluster.

11. Test the Web application installation as follows:

    a. In a Web browser, navigate to the Oracle Role Manager Web application address. For example:

```
http://localhost:9080/webui
```

    b. Log in as the Oracle Role Manager Administrator created in Section 3.3, "Installing Role Manager."

    You should see the Home page of the Oracle Role Manager Web application.

6

Configuring JBoss

This chapter includes the following sections:

- Configuring JBoss Server in a Nonclustered Mode
- Configuring JBoss in a Clustered Mode
- Encrypting the Role Manager Database Password

6.1 Configuring JBoss Server in a Nonclustered Mode

This procedure assumes that JBoss is installed on the application server host for Role Manager.

You must configure JBoss server in SSL mode to operate in a secure environment. For information about configuring SSL for JBoss server, refer to the following URL:

http://docs.jboss.org/jbossas/guides/webguide/r2/en/html_single/#ch9.https.sec

To configure JBoss for Role Manager

1. Copy the `orm-ds.xml` and `orm-service.xml` files from:

   ORM_HOME/samples/jboss/4.2.3

   to the JBoss server where you want to deploy Role Manager. For example:

   JBOSS_HOME/server/default/deploy

2. Set the session ID to false in the following path:

   JBOSS_HOME/server/default/deploy/jboss-web.deployer/server.xml file
   setting emptySessionPath="false" for the HTTP/1.1 Connector

   For example:

   ```
   <Connector port="8080" address="${jboss.bind.address}"
           maxThreads="250" maxHttpHeaderSize="8192"
           emptySessionPath="true" protocol="HTTP/1.1"
           enableLookups="false" redirectPort="8443" acceptCount="100"
           connectionTimeout="20000" disableUploadTimeout="true" />
   ```

3. Set the JTA transaction timeout parameter by performing the following steps:

   a. Open the `jboss-service.xml` file from the following path:

      JBOSS_HOME/server/default/conf/jboss-service.xml file

b. Locate the configuration for mbean with the name, "mbean code="com.arjuna.ats.jbossatx.jta.TransactionManagerService".

c. Change the `TransactionTimeout` attribute value to 1200:

```
<attribute name="TransactionTimeout">1200</attribute>
```

> **Note:** For more information about the JBoss Application Server, refer to the following link:
>
> http://www.jboss.org/docs/

4. Encrypt the password of the Role Manager application user defined in Section 6.3.

> **Note:** This step 4 is optional. Perform this step for development or non-production environments, where password encryption is not needed.

5. Edit the `orm-ds.xml` file as follows:

a. Change the two occurrences of `connection-url` to match your database environment:

```
<connection-url>jdbc:oracle:thin:@//SERVER_NAME:PORT/SERVICE_NAME</connection-url>
```

b. Change the two occurrences of `user-name` and `password` to match the credentials of the Role Manager application user and password:

```
<user-name>USER_NAME</user-name>
<password>PASSWORD</password>
```

6. Copy the server.ear file from *ORM_HOME*/lib to the JBoss directory used above.

7. Copy the webui.war file from *ORM_HOME*/webui/jboss/4.2.3 to the JBoss directory used above.

8. If JBoss is not already running, start the JBoss server using the following command:

For UNIX:

```
JBOSS_HOME/bin/run.sh
```

For Windows:

```
JBOSS_HOME\bin\run.bat
```

9. To test the server installation, ensure that you can get to the Role Manager administrative console from a Web browser. For example:

```
http://localhost:8080/ormconsole
```

10. To test the Role Manager Web application installation:

a. In a Web browser, navigate to the Role Manager Web UI. For example:

```
http://localhost:8080/webui
```

b. Log in as the Role Manager Administrator created in Section 3.3, "Installing Role Manager."

You should see the Home page of the Role Manager Web application.

Note: Data must be loaded into the system to expose all the functionality of the application. Refer to Section 7.2, "Loading Sample Data," for instructions.

6.2 Configuring JBoss in a Clustered Mode

This section includes the following topics:

- Installing Oracle Role Manager
- Configuring JBoss Server on the First Node
- Setting Up the Network
- Copying Oracle Role Manager to Additional JBoss Application Server Nodes
- Setting Up JMS on JBoss
- Modifying server.ear
- Creating finalization-server.ear
- Modifying webui.war
- Starting the JBoss Application Server on the First Node
- Copying and Starting Additional JBoss Server Nodes

Caution: Deploying an application in a clustered installation is a complex procedure. This document assumes that you have expertise in installing and using applications in a JBoss Application Server cluster. These instructions provide the Oracle Role Manager-specific details only. They are not complete instructions for setting up a JBoss Application Server cluster. For more information about clustering, see JBoss Application Server documentation.

6.2.1 Installing Oracle Role Manager

Follow the installation steps for Oracle Role Manager in Section 3.3, "Installing Role Manager," or Section 3.4, "Performing a Silent Installation Using a Response File" to install Oracle Role Manager.

6.2.2 Configuring JBoss Server on the First Node

To configure JBoss server on the first node:

1. Copy the `orm-service.xml` file from *ORM_HOME*/samples/jboss/4.2.3 to *JBOSS_HOME*/server/all/deploy-hasingleton/jms.

2. Copy the `orm-ds.xml` file from *ORM_HOME*/samples/jboss/4.2.3 to *JBOSS_HOME*/server/all/farm.

3. Edit the `orm-ds.xml` file as follows:

   a. Change the two occurrences of connection-url to match your database environment:

   <connection-url>jdbc:oracle:thin:@//SERVER_NAME:PORT/SERVICE_NAME</connection-url>

   b. Change the two occurrences of user name and password to match the credentials of the Role Manager application user with the newly encrypted password:

```
<user-name>USER_NAME</user-name>

<password>PASSWORD</password>
```

4. Copy the `server.ear` file from *ORM_HOME*/lib to *JBOSS_HOME*/server/all/farm.

5. Copy the `webui.war` file from *ORM_HOME*/webui/jboss/4.2.3 to *JBOSS_HOME*/server/all/farm.

6. Configure Cluster Name to DefaultPartition on `jboss-service.xml` of *JBOSS_HOME*/server/all/deploy/jboss-web-cluster.sar/META-INF.

6.2.3 Setting Up the Network

To set up the network for cluster:

1. Use IPv4 to ensure that you specify *-Djava.net.preferIPv4Stack=true*, otherwise JGroups might not start up.

2. Explicitly specify the network interface. On servers with multiple NICs, ensure that you select a specific NIC, otherwise JGroups may not be able to select a unique address for each node in the cluster.

3. Check firewalls to ensure firewalls and switches allow multi-cast IP on the server LAN.

4. Check for proper broadcast address to ensure the broadcast address is set properly on the NICs. Broadcast addresses are generally, x.y.z.255 where x.y.z is the subnet address.

5. All NICs used for the JBoss cluster should have the same broadcast address.

6.2.4 Copying Oracle Role Manager to Additional JBoss Application Server Nodes

> **Note:** You must ensure that the name and path of the *JAVA_HOME* directory used by Oracle Role Manager is same across all nodes of the cluster. For each additional node in your JBoss Application Server cluster, copy the JBoss and Oracle Role Manager installation directories from the first node to all other nodes, ensuring to maintain the original directory structure and hierarchy throughout this process.

6.2.4.1 TCP Based Clustering

JBoss can be configured to use TCP or UDP based clustering. By default, JBoss is configured for UDP. For more information about UDP based clustering, refer to the JBoss Clustering guide available at:
http://www.redhat.com/docs/en-US/JBoss_Enterprise_Application_Platform/

TCP uses unicast messages to communicate with other nodes whereas UDP uses multicast messages over the network. If there are more than 2 nodes in the cluster then TCP would cause increased network traffic. If your internal network policy does not allow UDP, then TCP is needed. If you are on Windows, change UDP element's attribute `loopback` to `true`.

For using TCP based clustering to automatically detect a node in a multi-homed JBoss setup, modify the configuration file, `cluster-service.xml` located in *JBOSS_HOME*/server/all/deploy as follows:

1. Comment out the following UDP section to use a multicast group for the cluster communication:

```
<!--
<Config>
   <UDP mcast_addr="${jboss.partition.udpGroup:228.1.2.3}" mcast_port="45566"
ip_ttl="${jgroups.mcast.ip_ttl:8}" ip_mcast="true"
mcast_recv_buf_size="2000000" mcast_send_buf_size="640000"
ucast_recv_buf_size="2000000" ucast_send_buf_size="640000" loopback="true"/>
            <PING timeout="2000" num_initial_members="3" up_thread="true"
down_thread="true"/>
            <MERGE2 min_interval="10000" max_interval="20000"/>
            <FD_SOCK down_thread="false" up_thread="false"/>
            <FD shun="true" up_thread="true" down_thread="true" timeout="10000"
max_tries="5"/>
            <VERIFY_SUSPECT timeout="3000" num_msgs="3" up_thread="true"
down_thread="true"/>
            <pbcast.NAKACK gc_lag="50"
retransmit_timeout="300,600,1200,2400,4800" max_xmit_size="8192"
up_thread="true" down_thread="true"/>
            <UNICAST timeout="300,600,1200,2400,4800" window_size="100"
min_threshold="10" down_thread="true"/>
            <pbcast.STABLE desired_avg_gossip="20000" max_bytes="400000"
up_thread="true" down_thread="true"/>
            <FRAG frag_size="8192" down_thread="true" up_thread="true"/>
            <pbcast.GMS join_timeout="5000" join_retry_timeout="2000"
shun="true" print_local_addr="true"/>
            <pbcast.STATE_TRANSFER up_thread="true" down_thread="true"/>
</Config>
-->
```

2. Uncomment the following section to use a TCP stack:

```
<Config>
   <TCP bind_addr="THISHOST" start_port="7800" loopback="true"
recv_buf_size="2000000" send_buf_size="640000" tcp_nodelay="true"
up_thread="false" down_thread="false"/>
   <TCPPING initial_hosts="THISHOST[7800],OTHERHOST[7800]" port_range="3"
timeout="3500" num_initial_members="3" up_thread="false" down_thread="false"/>
   <MERGE2 min_interval="5000" max_interval="10000" up_thread="false"
down_thread="false"/>
   <FD_SOCK down_thread="false" up_thread="false"/>
   <FD shun="true" up_thread="false" down_thread="false" timeout="10000"
max_tries="5"/>
   <VERIFY_SUSPECT timeout="1500" down_thread="false" up_thread="false" />
   <pbcast.NAKACK up_thread="false" down_thread="false" gc_lag="100"
retransmit_timeout="300,600,1200,2400,4800"/>
   <pbcast.STABLE desired_avg_gossip="20000" max_bytes="400000"
down_thread="false" up_thread="false" />
   <pbcast.GMS join_timeout="5000" join_retry_timeout="2000" shun="true"
print_local_addr="true"  up_thread="false" down_thread="false"/>
```

```
   <FC max_credits="2000000" down_thread="false" up_thread="false"
min_threshold="0.10"/>
   <FRAG2 frag_size="60000" down_thread="false" up_thread="true"/>
   <pbcast.STATE_TRANSFER up_thread="false" down_thread="false"/>
</Config>
```

3. Replace THISHOST and OTHERHOST with the IP addresses of the hosts in the cluster.

4. Add additional OTHERNODE entries as needed to support additional cluster members.

5. If you are using TCP, then change the 3 multi cast ports from 7800 to 45777 (this port change avoids warnings about discarded message from another partition, because the Tomcat-DefaultPartition now uses another set of ports).

6. When you copy cluster-service.xml to *JBOSS_HOME*/server/all/deploy on other cluster nodes, remember to modify THISHOST with the IP address of the node.

6.2.5 Setting Up JMS on JBoss

To set up JMS on JBoss:

1. Copy the *ORM_HOME*/lib/ojdbc14.jar file to *JBOSS_HOME/server/all/lib*.

2. Update the JBoss Persistence Manager service by performing the following steps.

   a. Copy *JBOSS_HOME*/docs/examples/jms/oracle-jdbc2-service.xml to *JBOSS_HOME*/server/all/deploy-hasingleton/jms.

   b. Modify the oracle-jdbc2-service.xml file to provide DefaultDS as the value for PersistenceManager:

   ```
   <depends
   optional-attribute-name="ConnectionManager">jboss.jca:service=DataSourceBin
   ding,name=DefaultDS</depends>
   ```

   c. Delete the deploy-hasingleton/jms/hsqldb-jdbc2-service.xml file

   d. Delete the *JBOSS_HOME*/server/all/farm/hsqldb-ds.xml file.

3. Deploy the data source descriptor for your database by renaming *JBOSS_HOME*/server/all/deploy-hasingleton/jms/hsqldb-jdbc-state-serv ice.xml to oracle-jdbc-state-service.xml and provide DefaultDS as the value for Connection Manager:

   ```
   <depends
   optional-attribute-name="ConnectionManager">jboss.jca:service=DataSourceBinding
   ,name=DefaultDS</depends>
   ```

4. Navigate to *JBOSS_HOME*/server/all/deploy/jms, open hajndi-jms-ds.xml and change the bind address to IP or DNS hostname of the machine.

   For Example:

   ```
   java.naming.provider.url=${jboss.bind.address:192.168.1.25}:1100
   ```

5. Navigate to *JBOSS_HOME*/server/all/farm, open orm-ds.xml and include the following block and change the url, username, password at two places:

   ```
   <no-tx-datasource>
     <jndi-name>DefaultDS</jndi-name>
   ```

```
    <!-- Oracle version: replace SERVER_NAME and SERVICE_NAME -->
    <driver-class>oracle.jdbc.OracleDriver</driver-class>
    <connection-url>jdbc:oracle:thin:@//
SERVER_NAME:port_no/service_name</connection-url>

<exception-sorter-class-name>org.jboss.resource.adapter.jdbc.vendor.OracleExcep
tionSorter</exception-sorter-class-name>

    <!-- Authentication: replace USER_NAME and PASSWORD -->
    <user-name>username</user-name>
    <password>password</password>
    <!-- To set the database authentication in a secure form (i.e. not clear
text)
            refer to the Oracle Role Manager installation guide. -->

    <check-valid-connection-sql>select 1 from
model</check-valid-connection-sql>

<!-- The minimum connections in a pool/sub-pool. Pools are lazily constructed
on first use -->
        <min-pool-size>5</min-pool-size>

        <!-- The maximum connections in a pool/sub-pool -->
        <max-pool-size>50</max-pool-size>

        <!-- this will be run before a managed connection is removed from the
pool for use by a client-->
        <check-valid-connection-sql>select count(model_id) from
model</check-valid-connection-sql>
      <metadata>
        <type-mapping>Oracle9i</type-mapping>
      </metadata>
 </no-tx-datasource>

<local-tx-datasource>
        <jndi-name>DefaultXADS</jndi-name>

        <!-- Oracle version: replace SERVER_NAME and SERVICE_NAME -->
        <driver-class>oracle.jdbc.OracleDriver</driver-class>
        <connection-url>jdbc:oracle:thin://
SERVER_NAME:port_no/service_name</connection-url>

<exception-sorter-class-name>org.jboss.resource.adapter.jdbc.vendor.OracleExcep
tionSorter</exception-sorter-class-name>

        <!-- Authentication: replace USER_NAME and PASSWORD -->
        <user-name>username</user-name>
        <password>password</password>
        <!-- To set the database authentication in a secure form (i.e. not
clear text)
            refer to the Oracle Role Manager installation guide. -->

        <check-valid-connection-sql>select 1 from
model</check-valid-connection-sql>

<!-- The minimum connections in a pool/sub-pool. Pools are lazily constructed
on first use -->
        <min-pool-size>5</min-pool-size>

        <!-- The maximum connections in a pool/sub-pool -->
```

```
            <max-pool-size>50</max-pool-size>
            <!-- this will be run before a managed connection is removed from the
pool for use by a client-->
            <check-valid-connection-sql>select count(model_id) from
model</check-valid-connection-sql>
        <metadata>
            <type-mapping>Oracle9i</type-mapping>
        </metadata>
    </local-tx-datasource>
```

6.2.6 Modifying server.ear

The `server.ear` file, supplied in the lib directory in the Oracle Role Manager installation directory needs the additional files and modifications to configuration files in order to work properly in a clustered server environment.

To modify server.ear:

1. Using a utility such as WinZip or jar, extract the contents of `server.ear` (located *ORM_HOME*/lib) into temporary directory.

    Within the extracted server.ear directory, you should see two additional application archives, `server.jar` and `ormconsole.war`.

2. Extract the contents of `server.jar` and `ormconsole.war` into two different temporary directories.

3. Modify the contents of `server.jar` as follows:

    a. Navigate to the root directory of the expanded `server.jar` in the temporary location.

      You should see two directories, `META-INF` and `oracle`.

    b. In this root directory, create a file named `jndi.properties` file with the following contents:

    ```
    java.naming.factory.initial=org.jnp.interfaces.NamingContextFactory
    java.naming.factory.url.pkgs=org.jboss.naming:org.jnp.interfaces
    java.naming.provider.url=jnp://orm_node1_ip_address:1100,orm_node2_ip_addre
    ss:1100/
    ```

    Note: If JBoss is getting bound to the DNS name, use the DNS name of the node instead of the IP address.

    ---

    c. In the META-INF directory open the `ejb-jar.xml` file with a text editor and remove the following four bean and transaction elements for BtFinisherEJB and BtFinisherMessageEJB:

    ```
    <entity id="Entity_1183672362011">
    ...
        <display-name>Finalization Server Bean</display-name>
        <ejb-name>BtFinisherEJB</ejb-name>
    ...
    </entity>

    <message-driven id="MessageDriven_1183672362010">
        <description>Message bean for handling incoming business transaction
    finalizations</description>
        <display-name>Finalization Server Message Bean</display-name>
        <ejb-name>BtFinisherMessageEJB</ejb-name>
    ```

```
...
</message-driven>

<method>
    <ejb-name>BtFinisherEJB</ejb-name>
    <method-name>*</method-name>
</method>

<method>
    <ejb-name>BtFinisherMessageEJB</ejb-name>
    <method-name>*</method-name>
</method>
```

d. In the same location, open the `jboss.xml` file with a text editor and remove the following entries for BtFinisherEJB and BtFinisherMessageEJB:

```
<entity>
    <ejb-name>BtFinisherEJB</ejb-name>
    <jndi-name>ejb/BtFinisher</jndi-name>
</entity>
<message-driven>
    <ejb-name>BtFinisherMessageEJB</ejb-name>
    <destination-jndi-name>queue/orm/BtFinisherQueue</destination-jndi-name
>
    <configuration-name>Singleton MDB</configuration-name>
</message-driven>
```

e. Repackage the contents of `server.jar`.

The directory layout in `server.jar` should include the added file as follows:

```
jndi.properties
META-INF/
oracle/
```

4. Modify the contents of `ormconsole.war` as follows:

a. Navigate to the `WEB-INF` directory of the expanded `ormconsole.war` in the temporary location.

b. In the WEB-INF directory, create a file named `jboss-web.xml` with the following content:

```
<?xml version='1.0' encoding='UTF-8' ?>
<!DOCTYPE jboss-web PUBLIC "-//JBoss//DTD Web Application 2.3V2//EN"
"http://www.jboss.org/j2ee/dtd/jboss-web_3_2.dtd">
<jboss-web>
  <replication-config>
    <replication-trigger>SET_AND_NON_PRIMITIVE_GET</replication-trigger>
    <replication-granularity>SESSION</replication-granularity>
    <replication-field-batch-mode>true</replication-field-batch-mode>
  </replication-config>
</jboss-web>
```

c. Create a subdirectory in the WEB-INF directory named `classes`.

d. Copy the `jndi.properties` file created in Step 3b into the newly created `classes` directory.

e. Repackage the contents of `ormconsole.war`.

The directory layout in `ormconsole.war` should include the added files as follows:

```
        login/
        META-INF/
        styles/
        WEB-INF/
            jboss-web.xml
            weblogic.wml
            web.xml
            ibm-web-bnd.xmi
            classes/
                jndi.properties
```

5. Repackage the contents of `server.ear`, ensuring to include the both updated `server.jar` and `ormconsole.war` files.

   The directory layout in `server.ear` should be as follows:

   ```
   META-INF/
   thirdparty/
   ormconsole.war
   server.jar
   ```

6. Copy `server.ear` to *JBOSS_HOME*/server/all/farm on all cluster nodes.

6.2.7 Creating finalization-server.ear

The `finalization-server.ear` file, must be created, configured ,and deployed as an additional archive in order to work properly in a clustered server environment.

To create finalization-server.ear:

1. Make a copy of the repackaged server.ear file and name it `finalization-server.ear`.

2. Using a utility such as WinZip or jar, extract the contents of `finalization-server.ear` into temporary directory.

   Within the extracted finalization-server.ear directory, you should see the `server.jar` file.

3. Navigate to the root directory of the expanded `finalization-server.jar` in the temporary location.

   You should see two directories, `META-INF` and `oracle`.

4. In the META-INF directory open the `ejb-jar.xml` file with a text editor and edit the file to contain only the content as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<ejb-jar version="2.1" id="EJBJar_1183672362010"
      xmlns="http://java.sun.com/xml/ns/j2ee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
http://java.sun.com/xml/ns/j2ee/ejb-jar_2_1.xsd">
   <enterprise-beans>
      <session id="Session_1183672362010">
         <description>The single source for the SubsystemRegistry for use by
other beans</description>
            <display-name>Singleton Bean</display-name>
            <ejb-name>SingletonEJB</ejb-name>
            <local-home>oracle.iam.rm.server_api.ejb.SingletonLocalHome</local-
home>
            <local>oracle.iam.rm.server_api.ejb.SingletonLocal</local>
            <ejb-class>oracle.iam.rm.server_api.ejb.SingletonBean</ejb-class>
```

```
                        <session-type>Stateless</session-type>
                        <transaction-type>Container</transaction-type>
                        <resource-ref id="ResourceRef_118367236209">
                                <res-ref-name>jdbc/server</res-ref-name>
                                <res-type>javax.sql.DataSource</res-type>
                                <res-auth>Container</res-auth>
                                <res-sharing-scope>Shareable</res-sharing-scope>
                        </resource-ref>
                        <resource-ref id="ResourceRef_1183672362010">
                                <res-ref-name>jdbc/serverxa</res-ref-name>
                                <res-type>javax.sql.DataSource</res-type>
                                <res-auth>Container</res-auth>
                                <res-sharing-scope>Shareable</res-sharing-scope>
                        </resource-ref>
                        <resource-ref id="ResourceRef_1183672362012">
                                <res-ref-name>jms/topicConFac</res-ref-name>
                                <res-type>javax.jms.TopicConnectionFactory</res-type>
                                <res-auth>Container</res-auth>
                                <res-sharing-scope>Shareable</res-sharing-scope>
                        </resource-ref>
                        <resource-ref id="ResourceRef_1183672362013">
                            <res-ref-name>jms/queueConnectionFactory</res-ref-name>
                            <res-type>javax.jms.QueueConnectionFactory</res-type>
                            <res-auth>Container</res-auth>
                            <res-sharing-scope>Shareable</res-sharing-scope>
                        </resource-ref>
                        <resource-ref id="ResourceRef_1183672362014">

<res-ref-name>ejb/BtFinisherConnectionFactory</res-ref-name>
                            <res-type>javax.jms.QueueConnectionFactory</res-type>
                            <res-auth>Container</res-auth>
                            <res-sharing-scope>Shareable</res-sharing-scope>
                        </resource-ref>
                        <resource-ref id="ResourceRef_1183672362015">
                                <res-ref-name>ejb/BtFinisherQueue</res-ref-name>
                                <res-type>javax.jms.Queue</res-type>
                                <res-auth>Container</res-auth>
                                <res-sharing-scope>Shareable</res-sharing-scope>
                        </resource-ref>
                        <resource-env-ref id="ResourceEnvRef_1183672362010">

<resource-env-ref-name>jms/loaderQueue</resource-env-ref-name>

<resource-env-ref-type>javax.jms.Queue</resource-env-ref-type>
                        </resource-env-ref>
                        <resource-env-ref id="ResourceEnvRef_1183672362011">

<resource-env-ref-name>jms/notification</resource-env-ref-name>

<resource-env-ref-type>javax.jms.Topic</resource-env-ref-type>
                        </resource-env-ref>
                        <security-identity>
                                <description>The role to use for managing finalization
server message delivery.</description>
                                <run-as>
                                        <role-name>ORMServer</role-name>
                                </run-as>
                        </security-identity>
                </session>
            <entity id="Entity_1183672362011">
```

```
            <description>An Entity Bean that is used to manage the "single commit
server" model.  Only one should ever exist at one time in a
cluster.</description>
            <display-name>Finalization Server Bean</display-name>
            <ejb-name>BtFinisherEJB</ejb-name>

<local-home>oracle.iam.rm.temporal.impl.ejb.BtFinisherHome</local-home>
            <local>oracle.iam.rm.temporal.impl.ejb.BtFinisher</local>
            <ejb-class>oracle.iam.rm.temporal.impl.ejb.BtFinisherBean</ejb-class>
            <persistence-type>Bean</persistence-type>
            <prim-key-class>java.lang.String</prim-key-class>
            <reentrant>false</reentrant>
            <ejb-local-ref id="EJBLocalRef_1183672362014">
                    <ejb-ref-name>ejb/singleton</ejb-ref-name>
                <ejb-ref-type>Session</ejb-ref-type>

<local-home>oracle.iam.rm.server_api.ejb.SingletonLocalHome</local-home>
                <local>oracle.iam.rm.server_api.ejb.SingletonLocal</local>
                <ejb-link>SingletonEJB</ejb-link>
            </ejb-local-ref>
        </entity>
          <message-driven id="MessageDriven_1183672362010">
            <description>Message bean for handling incoming business transaction
finalizations</description>
            <display-name>Finalization Server Message Bean</display-name>
            <ejb-name>BtFinisherMessageEJB</ejb-name>

<ejb-class>oracle.iam.rm.temporal.impl.ejb.BtFinisherMessageBean</ejb-class>
            <transaction-type>Container</transaction-type>
            <message-destination-type>javax.jms.Queue</message-destination-type>
            <ejb-local-ref id="EjbLocalRef_1183672362012">
                <ejb-ref-name>ejb/BtFinisher</ejb-ref-name>
                <ejb-ref-type>Entity</ejb-ref-type>

<local-home>oracle.iam.rm.temporal.impl.ejb.BtFinisherHome</local-home>
                <local>oracle.iam.rm.temporal.impl.ejb.BtFinisher</local>
                <ejb-link>BtFinisherEJB</ejb-link>
            </ejb-local-ref>
            <ejb-local-ref id="EJBLocalRef_1183672362015">
                <ejb-ref-name>ejb/singleton</ejb-ref-name>
                <ejb-ref-type>Session</ejb-ref-type>

<local-home>oracle.iam.rm.server_api.ejb.SingletonLocalHome</local-home>
                <local>oracle.iam.rm.server_api.ejb.SingletonLocal</local>
                <ejb-link>SingletonEJB</ejb-link>
            </ejb-local-ref>
        </message-driven>
          <assembly-descriptor>
         <security-role>
                <description>The role used for internal server management
processes.</description>
                <role-name>ORMServer</role-name>
         </security-role>
          <container-transaction>
             <method>
              <ejb-name>SingletonEJB</ejb-name>
              <method-name>*</method-name>
          </method>
                <trans-attribute>NotSupported</trans-attribute>
          </container-transaction>
```

```xml
        <container-transaction>
            <method>
          <ejb-name>BtFinisherEJB</ejb-name>
          <method-name>*</method-name>
        </method>
        <method>
          <ejb-name>BtFinisherMessageEJB</ejb-name>
          <method-name>*</method-name>
        </method>
        <trans-attribute>Required</trans-attribute>
        </container-transaction>
        </assembly-descriptor>
      </enterprise-beans>
 </ejb-jar>
```

5. In the same location, open the `jboss.xml` file with a text editor and edit the file to contain only the content as follows:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE jboss PUBLIC
    "-//JBoss//DTD JBOSS 4.0//EN"
    "http://www.jboss.org/j2ee/dtd/jboss_4_0.dtd">
<jboss>
    <enterprise-beans>
            <session>
            <ejb-name>SingletonEJB</ejb-name>
            <configuration-name>Singleton Stateless Session
Bean</configuration-name>
            <resource-ref>
                <res-ref-name>jdbc/server</res-ref-name>
                <jndi-name>java:/ORMServerDS</jndi-name>
            </resource-ref>
            <resource-ref>
                <res-ref-name>jdbc/serverxa</res-ref-name>
                <jndi-name>java:/ORMServerXADS</jndi-name>
            </resource-ref>
            <resource-ref>
                <res-ref-name>jms/topicConFac</res-ref-name>
                <jndi-name>java:/JmsXA</jndi-name>
            </resource-ref>
            <resource-ref>
                <res-ref-name>jms/queueConnectionFactory</res-ref-name>
                <jndi-name>java:/JmsXA</jndi-name>
            </resource-ref>
            <resource-ref>
                <res-ref-name>ejb/BtFinisherConnectionFactory</res-ref-name>
                <jndi-name>java:/JmsXA</jndi-name>
            </resource-ref>
            <resource-ref>
                <res-ref-name>ejb/BtFinisherQueue</res-ref-name>
                <jndi-name>queue/orm/BtFinisherQueue</jndi-name>
            </resource-ref>
            <resource-env-ref>
                <resource-env-ref-name>jms/notification</resource-env-ref-name>
                <jndi-name>topic/orm/NotificationTopic</jndi-name>
            </resource-env-ref>
            <resource-env-ref>
                <resource-env-ref-name>jms/loaderQueue</resource-env-ref-name>
                <jndi-name>queue/orm/LoaderQueue</jndi-name>
            </resource-env-ref>
                    <clustered>true</clustered>
```

```
                        <cluster-config>
                                <partition-name>ORM_CLUSTER</partition-name>
                        </cluster-config>
                </session>
                <entity>
                    <ejb-name>BtFinisherEJB</ejb-name>
                    <jndi-name>ejb/BtFinisher</jndi-name>
                </entity>
                <message-driven>
                    <ejb-name>BtFinisherMessageEJB</ejb-name>

<destination-jndi-name>queue/orm/BtFinisherQueue</destination-jndi-name>
                    <configuration-name>Singleton Message Driven
Bean</configuration-name>
                </message-driven>
            </enterprise-beans>
            <container-configurations>
              <container-configuration>
                    <container-name>Singleton Stateless Session Bean</container-name>
                    <call-logging>false</call-logging>

<invoker-proxy-binding-name>stateless-rmi-invoker</invoker-proxy-binding-name>
                    <container-interceptors>

<interceptor>org.jboss.ejb.plugins.ProxyFactoryFinderInterceptor</interceptor>
                        <interceptor>org.jboss.ejb.plugins.LogInterceptor</interceptor>

<interceptor>org.jboss.ejb.plugins.SecurityInterceptor</interceptor>
                        <!-- CMT -->
                        <interceptor
transaction="Container">org.jboss.ejb.plugins.TxInterceptorCMT</interceptor>
                        <interceptor
transaction="Container">org.jboss.ejb.plugins.CallValidationInterceptor</interc
eptor>
                        <interceptor
transaction="Container">org.jboss.ejb.plugins.StatelessSessionInstanceIntercept
or</interceptor>
                        <!-- BMT -->
                        <interceptor
transaction="Bean">org.jboss.ejb.plugins.StatelessSessionInstanceInterceptor</i
nterceptor>
                        <interceptor
transaction="Bean">org.jboss.ejb.plugins.TxInterceptorBMT</interceptor>
                        <interceptor
transaction="Bean">org.jboss.ejb.plugins.CallValidationInterceptor</interceptor
>

<interceptor>org.jboss.resource.connectionmanager.CachedConnectionInterceptor</
interceptor>
                    </container-interceptors>

<instance-pool>org.jboss.ejb.plugins.StatelessSessionInstancePool</instance-poo
l>
                    <instance-cache></instance-cache>
                    <persistence-manager></persistence-manager>
                    <container-pool-conf>
                        <MinimumSize>1</MinimumSize>
                        <MaximumSize>1</MaximumSize>
                    </container-pool-conf>
                </container-configuration>
```

```
        </container-configurations>
</jboss>
```

6. Repackage the contents of `server.jar`.

   The directory layout in `server.jar` should be as follows:

   ```
   jndi.properties
   META-INF/
   oracle/
   ```

7. Navigate up one directory and then to `finalization-server/META-INF`.

   You should see the application.xml file.

8. Open the `application.xml` file with a text editor and edit the file to contain only the content as follows:

   ```
   <?xml version="1.0" encoding="UTF-8"?>
   <application version="1.4" id="ORM_Finalization_Server"
               xmlns="http://java.sun.com/xml/ns/j2ee"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
   http://java.sun.com/xml/ns/j2ee/application_1_4.xsd">
       <display-name>ORM Finalization Server</display-name>
           <module id="server">
             <ejb>server.jar</ejb>
           </module>
   </application>
   ```

9. Repackage the contents of `finalization-server.ear`, ensuring to include the both updated `server.jar` and configuration files

   The directory layout in `finalization-server.ear` should be as follows:

   ```
   META-INF/
   thirdparty/
   server.jar
   ```

   Note: The file ormconsole.war must not be present in the finalization-server.ear file. If it is, remove it and repackage the EAR file.

   ---

10. Copy the new `finalization-server.ear` file to *JBOSS_HOME*/server/all/deploy-hasingleton on all cluster nodes.

6.2.8 Modifying webui.war

The `webui.war` supplied in the webui/jboss/4.2.3 directory in the Oracle Role Manager installation directory needs the following additional configuration to work properly in a cluster.

To modify webui.war:

1. Using a utility such as WinZip or jar, extract the contents of `webui.war` (located *ORM_HOME*/webui/jboss/4.2.3) into temporary directory.

   Within the extracted webui.war directory, you should see the WEB-INF directory.

2. In the WEB-INF directory, edit the `jboss-web.xml` file to add the replication configuration as shown in bold:

```
<?xml version='1.0' encoding='UTF-8' ?>
<!DOCTYPE jboss-web PUBLIC "-//JBoss//DTD Web Application 2.3V2//EN"
"http://www.jboss.org/j2ee/dtd/jboss-web_3_2.dtd">
<jboss-web>
  <ejb-ref>
    <ejb-ref-name>ejb/server</ejb-ref-name>
    <jndi-name>ejb/orm/ServerEJB</jndi-name>
  </ejb-ref>
  <replication-config>
    <replication-trigger>SET_AND_NON_PRIMITIVE_GET</replication-trigger>
    <replication-granularity>SESSION</replication-granularity>
    <replication-field-batch-mode>true</replication-field-batch-mode>
  </replication-config>
</jboss-web>
```

3. Create a subdirectory in the WEB-INF directory named `classes`.

4. In this classes directory, create a file named `jndi.properties` file with the following content:

```
java.naming.factory.initial=org.jnp.interfaces.NamingContextFactory
java.naming.factory.url.pkgs=org.jboss.naming:org.jnp.interfaces
java.naming.provider.url=jnp://orm_node1_ip_address:1100,orm_node2_ip_address:1
100/
```

> **Note:** If JBoss is getting bound to the DNS name, use the DNS name of the node instead of the IP address.

5. Repackage the `web.war` file, ensuring it has the same layout as the previous `web.war` file.

6. Copy `web.war` to *JBOSS_HOME*/server/all/farm on all cluster nodes.

6.2.9 Starting the JBoss Application Server on the First Node

1. Navigate to the directory *JBOSS_HOME*\bin, and then run the following command:

On Microsoft Windows:

```
run -c all -b <bind_address> -Djboss.partition.name=ORM_CLUSTER
```

On UNIX:

```
/run.sh -c all -b <bind_address> -Djboss.partition.name=ORM_CLUSTER
```

2. Access the Administration console by opening a browser and pointing it to the following URL and verify that you can successfully create and update IT Roles, Business Roles, or Organizations:

```
http://IP_address:portNo/webui
```

6.2.10 Copying and Starting Additional JBoss Server Nodes

> **Note:** You must ensure that the name and path of the *JAVA_HOME* directory used by Oracle Role Manager is same across all nodes of the cluster. For each additional node in your JBoss Application Server cluster, copy the JBoss and Oracle Role Manager installation directories from the first node to all other nodes, ensuring to maintain the original directory structure and hierarchy throughout this process.

To copy and start additional JBoss Server Nodes:

1. If using TCP, edit `cluster-service.xml` in *JBOSS_HOME*/server/all/deploy on each cluster node and replace `TCPPING initialhost` with the IP address of the node.

2. Edit the *JBOSS_HOME*/server/all/deploy/jms/`hajndi-jms-ds.xml` file and change the bind address to the IP Address or DNS hostname of the machine. For example:

    ```
    java.naming.provider.url=${jboss.bind.address:192.168.1.25}:1100
    ```

3. Start JBoss Application Server on each node using the similar command as of the first node.

6.3 Encrypting the Role Manager Database Password

This section describes how to encrypt the Role Manager database password in JBoss application server deployments. Specifically, you must perform the following steps to manually encrypt a password, and then modify the `orm-ds.xml` and login-config.xml files so that they can access the encrypted form of the password instead of the clear text version.

To encrypt the Role Manager database password:

1. Open a console window and navigate to the *JBOSS_HOME* directory.

2. Stop the JBoss server.

3. Run one of the following commands to encrypt the Role Manager database password. In this command, replace `password` with the actual password that you want to encrypt.

    For UNIX:

    ```
    java -cp
    "$JBOSS_HOME/lib/jboss-jmx.jar:$JBOSS_HOME/lib/jboss-common.jar:$JBOSS_HOME/ser
    ver/default/lib/jboss-jca.jar:$JBOSS_HOME/server/default/lib/jbosssx.jar"
    org.jboss.resource.security.SecureIdentityLoginModule password
    ```

    For Windows:

    ```
    java -cp
    "%JBOSS_HOME%/lib/jboss-jmx.jar;%JBOSS_HOME%/lib/jboss-common.jar;%JBOSS_HOME%
    /server/default/lib/jboss-jca.jar;%JBOSS_HOME%/server/default/lib/jbosssx.jar"
     org.jboss.resource.security.SecureIdentityLoginModule password
    ```

    This command returns an encoded form of the password you specify. For example, the password `Welcome1` is encoded as `3146f9cc50afd6a6df8592078de921bc`.

4. Highlight and copy the encoded password to paste later in the JBoss application policy element definitions.

5. Open the *JBOSS_HOME*/server/default/deploy/`orm-ds.xml` file in a text editor.

6. Delete the `<user-name>` and `<password>` elements from the `<no-tx-datasource>` element.

7. Add the following `<security-domain>` element to the end of the `<no-tx-datasource>` element:

```
<security-domain>EncryptDBPassword</security-domain>
```

8. Delete the `<user-name>` and `<password>` elements from the `<local-tx-datasource>` element.

9. Add the following `<security-domain>` element to the end of the `<local-tx-datasource>` element:

```
<security-domain>EncryptXADBPassword</security-domain>
```

10. Save and close the `orm-ds.xml` file.

11. Open the *JBOSS_HOME*/server/default/conf/`login-config.xml` file in a text editor.

12. Add the following to `<application-policy>` element at the end of the `<policy>` element while replacing *datasource_username* with the data source user name and *encoded_password* with the encoded password you copied in step 3:

```
<application-policy name = "EncryptXADBPassword">
 <authentication>
  <login-module code = "org.jboss.resource.security.SecureIdentityLoginModule"
flag = "required">
    <module-option name = "username">datasource_username</module-option>
    <module-option name = "password">encoded_password</module-option>
    <module-option name = "managedConnectionFactoryName">
      jboss.jca:service=LocalTxCM,name=ORMServerXADS</module-option>
  </login-module>
 </authentication>
</application-policy>

<application-policy name = "EncryptDBPassword">
 <authentication>
  <login-module code = "org.jboss.resource.security.SecureIdentityLoginModule"
flag = "required">
    <module-option name = "username">datasource_username</module-option>
    <module-option name = "password">encoded_password</module-option>
    <module-option name = "managedConnectionFactoryName">
jboss.jca:service=NoTxCM,name=ORMServerDS</module-option>
  </login-module>
 </authentication>
</application-policy>
```

13. Save and close the `login-config.xml` file.

7

Loading Data

This chapter provides the information about the following sections:

- Loading Standard Roles
- Loading Sample Data
- Manual Data Model Deployment
- Configuring the DAR File Size

7.1 Loading Standard Roles

The standard Roles contains objects and attributes that are listed in the standard data model, which are required for the Web application of Oracle Role Manager to function as designed. See Table A-1 in the Appendix of *Oracle Role Manager User's Guide* for a complete list of objects available in the standard data model of Oracle Role Manager.

To load standard roles:

1. If the application server on with Role Manager is deployed is not already running, start it.

2. In a Web browser, go to the Role Manager Administrative Console. For example, by default:

   JBoss: `http://<host>:8080/ormconsole`

   WebSphere: `http://<host>:9080/ormconsole`

   WebLogic: `http://<host>:9001/ormconsole`

3. Type user name and password of the Role Manager Administrator previously defined in "Installing Role Manager" on page 3-4, and then click **Log In**.

4. Click **Upload**.

5. Click **Browse**.

6. Navigate to select the `standard_roles.dar` file found in *ORM_HOME*`/samples/sample_data`.

7. Click **Load**.

8. Click **refresh** until you see that all processes are finalized.

7.2 Loading Sample Data

The sample data contains sample roles and role definitions, persons, and organizations. See Chapter 3, "Working with System Roles" of *Oracle Role Manager User's Guide* for more information about predefined system roles in the sample data.

To load standard and sample data:

1. If the application server on with Role Manager is deployed is not already running, start it.

2. In a Web browser, go to the Role Manager Administrative Console. For example, by default:

   JBoss: `http://<host>:8080/ormconsole`

   WebSphere: `http://<host>:9080/ormconsole`

   WebLogic: `http://<host>:9001/ormconsole`

3. Type user name and password of the Role Manager Administrator previously defined in "Installing Role Manager" on page 3-4, and then click **Log In**.

4. Click **Upload**.

5. Click **Browse**.

6. Navigate to select the `sample_data.dar` file found in *ORM_HOME*/`samples/sample_data`.

7. Click **Load**.

8. Click **refresh** until you see that all processes are finalized.

Once the data load processes display as being finalized, you can go back to the Role Manager Web application and check the sample data.

Loading the sample data can take several minutes. While data is being loaded, you can click **refresh** to monitor progress.

7.3 Manual Data Model Deployment

If you change the standard configuration or standard data model, you need to run a command to deploy your customizations to the database and then load the standard and the sample data.

This procedure assumes you have already completed the following steps:

- A database instance has been created for Role Manager with the appropriate tablespaces. (Refer to "Database Setup" on page 3-1.)

- The Role Manager database owner and application user schemas have been created and contain no data. (Refer to "Creating the Role Manager Users" on page 3-3.)

- The database is accessible and the service on which Role Manager is installed is started.

To deploy model and configuration customizations:

1. If you have any custom configuration or data model customizations, create an archive file containing your customizations and append the file name with `.car`. For more information about custom configuration, refer to *Oracle Role Manager*

Administrator's Guide and for more information about data model customization, refer to *Oracle Role Manager Developer's Guide*.

2. In *ORM_HOME*/`config`, if it does not exist, create a file named db.properties that contains the following two lines:

```
db.driverClass=oracle.jdbc.driver.OracleDriver
db.connection_string=jdbc:oracle:thin:@$HOST$:$PORT$:$SERVICE_NAME$
```

where:

- `$HOST$` is the database host name

- `$PORT$` is the database listener port

- `$SERVICE_NAME$` is the database instance on which the Role Manager users were created

3. In a command window, navigate to *ORM_HOME*/`bin`.

4. Run the following command to deploy the configuration and data model and create the Role Manager Administrator:

If you have no Integration Library (IL) customizations:

```
deploy.bat "../config/oim_integration.car" orm-owner ormapp-user admin-user
```

where:

- `orm-owner` is the user name of the Oracle Role Manager database owner user/schema

- `ormapp-user` is the user name of the Oracle Role Manager application user/schema

- `admin-user` is the user name of the Oracle Role Manager system administrator

If you have IL customizations:

```
deploy.bat "<collection_of_cars>" <orm-owner> <ormapp-user> <admin-user>
```

where:

- `<collection_of_cars>` contains the relative paths and file names of all CAR files to deploy.

  For example, in a customized deployment, the collection of CAR files on a UNIX-based system might be similar to:

  ```
  "../config/configurations_custom.car:../config/oim_integration_custom.car"
  ```

- `orm-owner` is the user name of the Oracle Role Manager database owner user/schema

- `ormapp-user` is the user name of the Oracle Role Manager application user/schema

- `admin-user` is the user name of the Oracle Role Manager system administrator

> **Note:** The collection must be enclosed within double quotation marks. The delimiters to be used are:
>
> - For Windows systems, use semicolon (;)
> - For UNIX-based systems, use a colon (:)

By default `standard.car` and `configurations.car` are part of initial deployment and if you have any customizations for these files, then you need to re-deploy customized files.

The first deployment must include the configuration.car* file. If you require the standard data model, for example, if you want to use the out-of-the-box web UI and/or the sample data, you must also include the standard.car file.

> **Note:** The default configuration.car file gives all privileges from the standard data model to the system administrator system role. If you want to start with a more hardened security policy, you must use the configuration_hardened.car file. You can use the hardened configuration and then decide to give privileges set in the default configuration.car file to the system administrator system role. To do so, you can either use the web UI or load the privilege mappings using the admin_systemrole_privilege_mapping.dar file.

<orm-owner> is the username of the Role Manager database owner user created in "Creating the Role Manager Users" on page 3-3

<ormapp-user> is the username of the Role Manager application user created in "Creating the Role Manager Users" on page 3-3

<admin-user> is the username for the Role Manager Administrator to create.

5. At the prompt, type the password of the Role Manager database owner.

6. At the prompt, type the password of the Role Manager application user.

7. At the prompt, type the password for the new Role Manager Administrator account.

> **Note:** After deploying the data model and configuration, you must load the sample and the standard data. Refer to "Loading Sample Data" on page 7-2 for instructions.

7.4 Configuring the DAR File Size

By default the largest DAR file you can load is 10MB, loading a larger file results in an error. You can configure the maximum DAR file size using the following instructions.

To configure the DAR file size:

1. Edit the config file:

   For UNIX-based systems:

   JBOSS_HOME/bin/run.sh

   For Windows systems:

JBOSS_HOME\bin\run.bat

2. Add the following argument to JAVA_OPTS:

`-Doracle.iam.rm.loader.max_upload_size=<new_value>`

8

Removing Oracle Role Manager Software

This chapter describes the process of removing Oracle Role Manager software. It contains the following sections:

- Removing Oracle Role Manager Software

  This section describes general instructions for UNIX-based systems and Windows systems.

- Removing the Oracle Role Manager Database

  This section describes specific instructions for dropping the Oracle Role Manager database users/schemas.

8.1 Removing Oracle Role Manager Software

Use the following procedure to uninstall the Oracle Role Manager software:

1. Run the Oracle Universal Installer as follows:

   For UNIX-based systems, run the command from $*ORACLE_HOME*/oui/bin:

   ```
   ./runInstaller.sh –deinstall -silent REMOVE_HOMES={"$ORACLE_HOME"}
   ```

   For example,

   ```
   $ORACLE_HOME/oui/bin/runInstaller.sh -deinstall -silent
   REMOVE_HOMES={"/scratch/ORMHome_1"}
   ```

   For Windows systems, run the following command from $*ORACLE_HOME*\oui\bin:

   ```
   setup.exe -deinstall -silent REMOVE_HOMES={"$ORACLE_HOME"}
   ```

   For example,

   ```
   ORACLE_HOME\oui\bin\setup.exe -deinstall -silent REMOVE_HOMES={"C:\orm123"}
   ```

2. As an alternative for Windows systems, start Oracle Universal Installer from the **Start** menu. On the Welcome page, click **Deinstall Products** to bring up the Oracle Inventory screen.

   Select the Oracle home directory and the products that you want to remove by selecting the desired check boxes, then click **Remove**. The Confirmation window appears. Click **Yes** to remove the selected components.

   After the Oracle Role Manager components are removed from your system, the **Oracle Inventory** page appears without the removed components. Click **Close** to

close the **Oracle Inventory** page. Click **Cancel** to exit Oracle Universal Installer. Click **Yes** to confirm that you want to exit.

3. Clean up the old Oracle directories.

   On systems where Oracle Role Manager is the only Oracle software installed, navigate to the directory for `oracle`, then remove the directory.

   For UNIX-based systems, use the `rm -r` command.

   Otherwise, delete the Oracle Role Manager home.

   For UNIX-based systems, issue the following command to confirm there is no other Oracle home installed.

   ```
   $ grep 'HOME NAME' OraInventory/ContentsXML/Inventory.xml
   ```

4. Remove the deployments of the Oracle Role Manager server and Web UI from the application server, as appropriate to the application server. For more information about cleaning up the application server deployments, refer to the corresponding application server documentation.

8.2 Removing the Oracle Role Manager Database

Use the following procedure to remove the Oracle Role Manager database:

1. As the Oracle SYSTEM user, use sqlplus or similar utility to perform the following tasks.

2. Ensure that there are no active sessions from users using the following commands:

   ```
   select sid,serial# from v$session where username = 'ORM_DB_OWNER';
   select sid,serial# from v$session where username = 'ORM_APP_USER';
   ```

3. Drop the Oracle Role Manager application user and the database owner using the following commands:

   ```
   drop user ORM_APP_USER cascade;
   drop user ORM_DB_OWNER cascade;
   ```

4. Drop the Oracle Role Manager tablespaces using the following commands.

   ```
   DROP TABLESPACE ORM_UNDO INCLUDING CONTENTS AND DATAFILES CASCADE
   CONSTRAINTS;
   DROP TABLESPACE ORM_TEMP INCLUDING CONTENTS AND DATAFILES CASCADE
   CONSTRAINTS;
   DROP TABLESPACE ORM_INDEX INCLUDING CONTENTS AND DATAFILES CASCADE
   CONSTRAINTS;
   DROP TABLESPACE ORM_DATA INCLUDING CONTENTS AND DATAFILES CASCADE
   CONSTRAINTS;
   commit;
   ```

9

Upgrading Oracle Role Manager

This chapter provides a detailed information about Oracle Role Manager upgrade process. It deals with upgrading to Oracle Role Manager release 10.1.4.2 from release 10.1.4.1 or release 10.1.4.1.1.

To upgrade to release 10.1.4.2 from release 10.1.4.1 or release 10.1.4.1.1, you must complete the following tasks:

- Installing the Oracle Role Manager in Install-Only Mode
- Creating a Back Up for Existing Database
- Configuring Oracle Role Manager Integration Library
- Re-applying Customization to Data Model Configurations
- Running the Upgrade Tool
- Upgrade Logging
- Verifying the Upgrade
- Re-Deploying New Binaries to J2EE Container

9.1 Installing the Oracle Role Manager in Install-Only Mode

You must run the installer in the install only mode. This is because Role Manager configuration is already existing and you are upgrading it to the latest configuration using the upgrade utility. Running the installer in the install only mode will deliver the latest files, configurations.car and standard.car and binaries, server.jar, server.ear, webui.ear to your environment. For information about running the Oracle Role Manager installer, refer "Installing Role Manager" on page 3-4.

The installer does not support overwriting an existing installation on the file system. You must provide an alterative file system path to the installer.

> **Note:** The install and configure mode is not supported on existing dbowner/app user schema which has data, but it is supported only on new dbowner/app user schema which has no data. Therefore, to upgrade existing configuration, you must install Oracle Role Manager in the install-only mode.

9.2 Creating a Back Up for Existing Database

Use the export or backup utilities provided with the database to perform a complete backup of your production database.

Production database backup includes, but is not limited to, complete export or backup of the Oracle Role Manager release 10.1.4.1 or 10.1.4.1.1 database instance to ensure that the database can be restored to its original state, if required.

> **Note:**
>
> ■ You must shut down the J2EE server in which Oracle Role Manager is running (or all servers in case of a cluster) before running the database backup and the upgrade tool.
>
> ■ Ensure that no other JDBC client is accessing the existing Oracle Role Manager database.

9.3 Configuring Oracle Role Manager Integration Library

If you have installed Oracle Role Manager Integration Library for Oracle Identity Manager, then you must configure the Integration Library to match the Oracle Role Manager version that you are going to install. For information about how to configure the Integration Library, refer to *Oracle Role Manager Integration Guide*

9.4 Re-applying Customization to Data Model Configurations

If Oracle Role Manager setup has any customization to the data model configuration files such as standard.xml or standard_permissions.xml, then you must reapply the customized configuration to the latest data model configurations shipped with Oracle Role Manager 10.1.4.2. For more information about data model customization, refer to *Oracle Role Manager Developer's Guide.*

> **Note:**
>
> ■ Re-applying customization is optional if the Oracle Role Manager setup does not have any customization to standard data model configurations and webui.
>
> ■ Upgrade does not support upgrade of webui and datamodel customizations. These customizations must be applied manually.

9.5 Running the Upgrade Tool

The upgrade tool is a command line tool, which upon successful running, upgrades the existing Oracle Role Manager database to the latest one.

The upgrade tool will not upgrade the Oracle Role Manager software deployed to application servers, but only upgrade the data model and configurations in the database.

> **Note:** You must not directly run the upgrade tool on production database. It must be run on staging environment, which is prepared from production database backup. Upon successful upgrade and acceptance testing in staging environment, this tool can be run on production database.

Running the upgrade tool provides the installation with the following details:

- Updated standard configurations

- Updated Oracle Role Manger Integration Library configurations

- Updated custom configurations

For more information about these configurations, refer to *Oracle Role Manager Integration Guide*.

To run the upgrade tool:

1. On the Oracle Role Manager installation host, navigate to *ORM_HOME*/config on the new ORM installation.

2. Ensure that the db.properties file in *ORM_HOME*/config contains the correct information for your database environment. If it does not, edit and add the following lines:

   ```
   db.driverClass=oracle.jdbc.driver.OracleDriver
   db.connection_string=jdbc:oracle:thin:@//$HOST$:$PORT$/$SERVICE$
   ```

   Where $HOST$ is the database host name, $PORT$ is the database listener port, and $SERVICE$ is the database instance on which the existing Oracle Role Manager users/schemas were created.

3. Stop the Oracle Role Manager application server if it is running.

4. In a command window, navigate to *ORM_HOME*/bin.

5. Run the following command to deploy the configuration:

   ```
   upgrade.sh NEW_ORM_HOME ..\config\db.properties ormapp-user orm-owner
   admin-user  "collection_of_cars"
   ```

   Where:

   - `ormapp-user` is the user name of the Oracle Role Manager application user/schema

   - `orm-owner` is the user name of the Oracle Role Manager database owner user/schema

   - `admin-user` is the user name of the Oracle Role Manager System Administrator

   - `collection_of_cars` contains the relative paths and file names of CAR files to be deployed. This collection must be within quotes with a semicolon (;) used as the delimiter between CAR files.

     Example for default configurations upgrade:

     ```
     "..\config\standard.car"
     ```

     Example for customized configurations upgrade:

     ```
     "..\model_custom\datamodel_custom.car;..\model_custom\standard_custom.car;"
     ```

     Example for a customized Oracle Role Manager configurations and Integration Library configurations upgrade:

     ```
     ..\config\standard_custom.car;..\config\oim_integration_custom.car
     ```

6. At the prompt, type Y to confirm you want to proceed with the upgrade.

7. At the prompt, type the passwords for application user, database owner, and administrator. The command window displays messages about the progress of the upgrade. These messages are also recorded in the log file.

   Refer "Upgrade Logging" on page 9-4 for examples of log messages.

   Refer "Verifying the Upgrade" on page 9-4 for information about verifying that the upgrade was successful.

9.6 Upgrade Logging

The log files for upgrade, for example `orm-upgrade-1.log`, are generated under the `logs` folder of *ORM_HOME*. These log files contain details about the progress of upgrade tasks and failure or success reports for upgrade.

The following are the examples of log messages generated in the log file:

- New connection for orm_db_owner

- Current detected ORM version is 10.1.4.1

- Starting the Upgrade Task, clean up any objects in a hierarchy with multiple parents

- Exception: Version compatibility check failed. Cannot proceed.

9.7 Verifying the Upgrade

The upgrade log will contain following messages that can be used to determine if the upgrade has been started and completed successfully:

- Starting upgrade from version 10.1.4.1 to version 10.1.4.2

- Version table created Successfully

- Version table populated with version number 10.1.4.2

- Upgrade Completed Successfully.

In case the upgrade fails, then an Upgrade Exception with the failure details will be logged in the log file.

Fix the error accordingly and rerun the upgrade till it runs successfully. If required restore the database from the production database backup.

If you complete the upgrade successfully, then a table named `VERSION_TABLE` is created in the `orm-owner` database schema and the `SOFTWARE_VERSION` column of the table displays the latest Oracle Role Manager version.

9.8 Re-Deploying New Binaries to J2EE Container

To re-deploy new binaries to J2EE container:

1. If the server.ear and the webui.war file are already deployed into the J2EE container, undeploy them and re-deploy the latest ones again.

2. Start up the J2EE container/application server that hosts the Oracle Role Manager instance.

   > **Note:** If IL is configured, then you must start the upgraded Oracle Identity Manager instances.

3. If IL is configured for Oracle Identity Manager, then perform the acceptance testing including IL. If acceptance test passes on staging environment, then repeat the steps from 9.6 to 9.8 on production environment.

9.8.1 For WebLogic

To deploy the Role Manager server and Web applications for WebLogic, refer "Deploying Role Manager" on page 4-6.

9.8.2 For JBoss

To deploy the Role Manager server and Web applications for JBoss, refer "Configuring JBoss Server in a Nonclustered Mode" on page 6-1.

9.8.3 For WebSphere

To deploy the Role Manager server and Web applications for WebSphere, refer "Deploying the Oracle Role Manager Web Application" on page 5-26 and "Deploying the Oracle Role Manager Server" on page 5-25.

A

Manual Configuration of WebLogic Server

This chapter provides the detailed explanation about manual configuration of WebLogic server. Perform these steps only if you intend to configure the WebLogic server manually.

A.1 Creating a New Domain

To create a new domain:

1. Start the Configuration Wizard using the following command:

    For Windows

    Go to Start, Oracle WebLogic, WebLogic Server 11gR1, Tools, Configuration Wizard.

    For UNIX:

    a. Go to the WebLogic bin directory using the following command:

    ```
    cd BEA_HOME/wlserver_10.3/common/bin
    ```

    b. Start the configuration wizard using the following command:

    ```
    sh config.sh
    ```

    Note: These instructions assume that the installation is a default installation performed with WebLogic server.

    ---

2. In the Configuration Wizard:

    a. Select **Create a New WebLogic domain** and then click **Next**.

    b. Select Base this domain on an existing template and choose WebLogic server template, for example, *<WLS_HOME>\common\templates\domains\wls.jar*. and click **Next**.

    c. The domain name and domain location are displayed by default. Click **Next**.

    d. Type the user name and the password, and then confirm the password for the domain. Click **Next**.

    e. On the left side of the window, select **Production Mode**.

> **Note:** If WebLogic server is configured in production mode, you need to restart the ORMServer to login.

    f. On the right side of the window, select the appropriate JDK and then click **Next**.

> **Note:** If you are using your own JDK, ensure that it is the certified JDK for Oracle WebLogic server.

    g. In Select Optional Configuration window, click **Next**.

    h. In Configuration Summary window, verify the details and then click **Create**.

    i. Click **Done**.

A.2 Configuring Memory options

To configure memory options:

- For Microsoft Windows systems

  Edit the startManagedWebLogic.cmd script and specify memory options as follows:

  Follow the format:

  %JAVA_HOME%\bin\java %JAVA_VM% %MEM_ARGS% %JAVA_OPTIONS%

  Locate the following line:

  ```
  set JAVA_OPTIONS=%JAVA_OPTIONS% -XnoOpt
  ```

  > **Note:**
  > - -XnoOpt is available only in JRockit.
  > - The -XnoOpt option turns off adaptive optimization and is required for stable Oracle Role Manager operation.

  Before this line, add the following if using Sun Java Virtual Machine (JVM):

  ```
  set MEM_ARGS=-Xms1280m -Xmx1280m -XX:PermSize=128m
  -XX:MaxPermSize=256m
  ```

  Add the following if using JRockit Java Virtual Machine (JVM):

  ```
  set MEM_ARGS=-Xms1280m -Xmx1280m
  ```

- For UNIX Systems

  Edit the startManagedWebLogic.sh script and specify memory options as follows:

  Locate the line that starts with the following:

  $*JAVA_HOME*/bin/java ${JAVA_VM} ${MEM_ARGS} ${JAVA_OPTIONS}

  If using Sun Java Virtual Machine (JVM), add the following line before this line:

  ```
  MEM_ARGS="-Xms1280m -Xmx1280m -XX:PermSize=128m
  -XX:MaxPermSize=256m"
  ```

```
export MEM_ARGS
```

If using JRockit Java Virtual Machine (JVM), add the following line before this line:

```
MEM_ARGS="-Xms1280m -Xmx1280m"
```

```
export MEM_ARGS
```

```
JAVA_OPTIONS="$JAVA_OPTIONS -XnoOpt"
```

```
export JAVA_OPTIONS
```

A.3 Starting WebLogic Server

To start the Oracle WebLogic server:

For Microsoft Windows:
1. For WebLogic, go to Start, Oracle WebLogic, User Projects, Domain Name, and then click **Start Admin Server for WebLogic** Server Domain.

2. Log on to the WebLogic Server Administration Console by using your new account and by pointing a Web browser to the following URL:

   ```
   http://hostname:7001/console
   ```

For Linux:
1. Go to the WebLogic user_projects/domains directory, for example:

   cd *BEA_HOME*/user_projects/domains/

2. Go to the directory of the domain that you just created using the Configuration Wizard. For example, cd domain name.

3. Start the Oracle WebLogic server using the following command:

   ```
   sh startWebLogic.sh
   ```

4. Log on to the WebLogic Server Administration Console by using your new account and by pointing a Web browser to the following URL:

   ```
   http://hostname:7001/console
   ```

A.4 Configuring Machines

A machine is a host that runs a WebLogic Server instance. You can configure a machine using the WebLogic Server Administration Console.

To configure a machine:
1. Select **Machines** under Environment and then click **New**.

2. In the **Name** field, type a name for the machine.

3. In the **Machine OS** field, select the operating system.

4. Click **OK**.

To start the node manager:
Use the following steps to start the node manager:

1. Navigate to `WebLogic_install_dir/server/bin`

2. Run the `startNodeManager.cmd` command for Microsoft Windows.

3. Run the `startNodeManager.sh` command for Linux.

A.5 Configuring Servers

> **Note:** WebLogic server must be already installed on which Role Manager application is configured.

The server is the machine that hosts the application. Use the following steps to configure a server in the WebLogic Server Administration Console:

To configure a server:

1. Select **Servers** under Environment and then click **New**.

2. In the **Server Name** field, type the name of the Role Manager server, for example `ORMServer`.

3. In the **Server Listen Address** field, type the IP address of the server.

4. In the **Server Listen Port** field, type the Role Manager listening port number .

5. Specify whether the server should belong to a cluster.

6. Click **Next**. Review your choices and then click **Finish**.

To assign a machine to the server:

1. Click the newly created server from the table of servers.

2. Click the **Configuration** tab and then click the **General** subtab.

3. In the **Machine** field, type the machine name.

4. Click **Save**.

To start the server:

1. Select **Servers** under Environment, and then click the newly created server from the table of servers.

2. Click the **Control** tab, then click the **Start/Stop** subtab.

3. Select **ORMServer** and then click **Start**.

A.6 Setting Up Commons Logging

Setting up Commons Logging for Oracle Role Manager on WebLogic involves creating a new Log4j appender, adding jars to ORM_WLS_DOMAIN_HOME/lib and adding JAVA_OPTION in startManagedWebLogic.cmd for Microsoft Windows and startManagedWebLogic.sh for Linux.

To Set Up Commons Logging:

1. Create a new file, log4j.properties in `ORM_WLS_DOMAIN_HOME` for example, *BEA_HOME*`/user_projects/domains/orm_domain`:

   ```
   log4j.rootLogger=debug, ORM

   log4j.logger.oracle.iam.rm=debug
   log4j.logger.oracle.iam.rm.persistence=INFO
   ```

```
log4j.logger.org.springframework=INFO
log4j.logger.org.apache=WARN
log4j.logger.org.quartz=WARN

# Uncomment all these stdout lines if logs are desired on console
#log4j.appender.stdout=org.apache.log4j.ConsoleAppender
#log4j.appender.stdout.layout=org.apache.log4j.PatternLayout
#log4j.appender.stdout.layout.ConversionPattern=%5p [%d] %t %c (%F:%L) - %m%n
#log4j.appender.stdout.threshold=info

log4j.appender.ORM=org.apache.log4j.RollingFileAppender
log4j.appender.ORM.File=C:/bea/user_projects/domains/orm_domain/servers/ORMServer/logs/ORM.log
# Change the threshold to INFO for less verbose logging
log4j.appender.ORM.threshold=debug
log4j.appender.ORM.MaxFileSize=1024KB
# Keep backup files
log4j.appender.ORM.MaxBackupIndex=5

log4j.appender.ORM.layout=org.apache.log4j.PatternLayout
log4j.appender.ORM.layout.ConversionPattern=%5p [%d] %t %c (%F:%L) - %m%n
```

2. In the file `ORM_WLS_DOMAIN_HOME/bin/startManagedWebLogic.cmd`, add a new line after line 55:

   `@REM`. Set `JAVA_OPTIONS` to the java flags that you want to pass to the vm.

   ```
   set JAVA_OPTIONS=%JAVA_OPTIONS%
   -Dlog4j.configuration=file:C:/bea/user_projects/domains/orm_domain/log4j.properties
   ```

   Replace "C:/bea" with *BEA_HOME*.

3. Add the following jars into `ORM_WLS_DOMAIN_HOME/lib` folder:

   - `commons-logging-1.0.4.jar` (download from apache.org)
   - `log4j-1.2.8.jar` (download from apache.org)

4. Restart ORM WebLogic server and you see ORM debug messages in the log-file:

   BEA_HOME`/user_projects/domains/orm_domain/servers/ORMServer/logs/ORMServer.log`

A.7 Configuring JDBC Data Sources

To configure the non-XA data source:

1. In the Domain tree, select Services, JDBC, Data Sources.

2. Click **New**. The JDBC Data Source Properties page appears.

3. In the **Name** field, type the name of the data source, for example, `ORM Data Source`.

4. In the **JNDI Name** field, type the JNDI path :

   `orm/jdbc/ORMServerDS`

5. From the Database Type list, select **Oracle**.

6. From the Database Driver list, select **Oracle's Driver (Thin)**.

7. Click **Next** and deselect **Supports Global Transactions**.

8. Click **Next**. The Connect Properties page appears.

9. In the **Database Name** field, type the name of the database, for example, (Oracle SID).

10. In the **Host Name** field, type the host name or IP address of the machine hosting the database.

11. In the **Port** field, type the port number on which the database is listening, for example, 1521.

12. In the **Database User Name** field, type the database user name that was created in the Step 3 of the "Creating the Role Manager Users" on page 3-3 section.

13. In the **Password** field, type the password for the database user and in the **Confirm Password** field, retype the password.

14. Click **Next**. The Test Database Connection page appears.

15. Verify the contents.

16. Click **Next**. The Select Targets page appears.

17. Select **ORMServer** as the target and then click **Finish**.

To configure the XA data source:

1. In the Domain tree, select Services, JDBC, Data Sources.

2. Click **New**. The JDBC Data Source Properties page appears.

3. In the **Name** field, type the name of the data source, for example, ORM XA Data Source.

4. In the **JNDI Name** field, type the JNDI path :

   orm/jdbc/ORMServerXADS

5. From the Database Type list, select **Oracle**.

6. From the Database Driver list, select Oracle's Driver (Thin XA).

7. Click **Next**.

8. The Transaction Options page appears. Click **Next**.

9. In the **Database Name** field, type the name of the database, for example, (Oracle SID).

10. In the **Host Name** field, type the host name or IP address of the machine hosting the database.

11. In the **Port** field, type the port number on which the database is listening, for example, 1521.

12. In the **Database User Name** field, type the database user name that was created in the Step 3 of the "Creating the Role Manager Users" on page 3-3 section.

13. In the **Password** field, type the password for the database user and in the Confirm Password field, retype the password.

14. Click **Next**. The Test Connection Database page appears.

15. Verify the contents.

16. Click **Next**. The Select Targets page appears.

17. Select **ORM server** as the target and then click **Finish**.

> **Note:** If you are using RAC database, provide the following string while creating the data source:
>
> ```
> jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP
> )(HOST=host1-vip)(PORT=1521))(ADDRESS=(PROTOCOL=TCP)(HOST=host2-vip
> )(PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=orcl.us.oracle.com)))
> ```

A.8 Configuring JMS Messaging Resources

To configure the JMS Server:

1. In the domain tree, select Services, Messaging, JMS Servers.

2. Click **New**. The JMS Server Properties page appears.

3. In the **Name** field, type the name of the Role Manager JMS Server, for example, `ORM JMSServer`.

4. Click **Next**. The Select targets page appears.

5. In the **Target** field, select **ORMServer** as the target and then click **Finish**.

To configure the JMS Module:

1. In the domain tree, select Services > Messaging > JMS Modules.

2. Click **New**. The Create JMS System Module page appears.

3. In the **Name** field, type the name of the Role Manager JMS module, for example, `ORM JMSModule`.

4. Click **Next**. The Target page appears.

5. In the **Target** field, select **ORMServer** as the target and then click **Next**.

6. Click **Finish**.

A.9 Configuring JMS Module Subdeployment

To configure JMS Module Subdeployment:

1. In the domain tree, select Services, Messaging, JMS Modules.

2. Click **ORM JMSModule**, for which you want to configure the subdeployment. Click the **Subdeployments** tab.

3. Click **New**. The Subdeployment Properties page appears.

4. In the **Name** field, type the name of the subdeployment, for example, `ORM JMSSubdeployment`.

5. Click **Next**. The Target page appears.

6. Select **ORM JMSServer** as the JMS server target and then click **Finish**.

A.10 Configuring the JMS Connection Factory

To configure a JMS Connection Factory:

1. In the domain tree, select Services, Messaging, JMS Modules.

2. Click **ORM JMSModule**.

3. Click **New** under Summary of Resources.

4. Select **Connection Factory** from the list of JMS resources.

5. Click **Next**. The Connection Factory Properties page appears.

6. In the **Name** field, type the name for the connection factory, for example, `ORM ConnectionFactory`.

7. In the **JNDI Name** field, type the path for the connection factory, for example, `orm/jms/ConnFac`.

8. Click **Next** and then click **Finish**.

9. In the domain tree, select Services, Messaging, JMS Modules.

10. Click **ORM JMSModule**.

11. Click **ORM ConnectionFactory**.

12. Navigate to the **Configuration** tab, **Transactions** sub-tab.

13. Select the **XA Connection Factory Enabled** check box.

14. Click **Save**.

To configure a Finalization Connection Factory:

1. In the domain tree, select Services, Messaging, JMS Modules.

2. Click **ORM JMSModule**.

3. Click **New** under Summary of Resources.

4. Select **Connection Factory** from the list of JMS resources.

5. Click **Next**. The Connection Factory Properties page appears.

6. In the **Name** field, type the name for the connection factory, for example, `Finalization ORM ConnectionFactory`.

7. In the **JNDI Name** field, type the path for the connection factory, for example, `orm/jms/FinalizationConFac`.

8. Click **Next**, and then click **Finish**.

9. In the domain tree, select Services, Messaging, JMS Modules.

10. Click **ORM JMSModule**.

11. Click **Finalization ORM ConnectionFactory**.

12. Navigate to the **Configuration** tab, **Transactions** sub-tab.

13. Select the **XA Connection Factory Enabled** check box.

14. Click **Save**.

A.11 Configuring a Notification Topic

To configure a notification topic:

1. In the domain tree, select Services, Messaging, JMS Modules.

2. Click **ORM JMSModule**.

3. Click **New** under Summary of Resources.

4. Select **Topic** from the list of JMS resources.

5. Click **Next**. The JMS Destination Properties page appears.

6. In the **Name** field. type a name for the topic, for example, `ORM NotificationTopic`.

7. In the **JNDI Name** field, type the path for accessing the topic, for example, `orm/topic/NotificationTopic`.

8. In the **Template** field, select **None**.

9. Click **Next**.

10. Select **ORM JMSSubdeployment** that you created in the Step 4 of the "Configuring JMS Module Subdeployment" on page A-7 section from the list.

11. Click **Finish**.

A.12 Configuring Queues

To configure the ORM Finisher Queue:

1. In the domain tree, select **Services, Messaging, JMS Modules**.

2. Click **ORM JMSModule**.

3. Click **New** under Summary of Resources.

4. Select **Queue** from the list of JMS resources.

5. Click **Next**. The JMS Destination Properties page appears.

6. In the **Name** field, type a name for the queue, for example, `ORM FinisherQueue`.

7. In the **JNDI Name** field, type the path for accessing the topic as `orm/queue/BtFinisherQueue`.

8. In the **Template** field, select **None**.

9. Click **Next**.

10. Select **ORM JMSSubdeployment** that you created in the Step 4 of the "Configuring JMS Module Subdeployment" on page A-7 section from the list.

11. Click **Finish**.

To configure the ORM Loader Queue:

1. In the domain tree, select **Services**, **Messaging**, **JMS Modules**.

2. Click **ORM JMSModule**.

3. Click **New** under Summary of Resources.

4. Select **Queue** from the list of JMS resources.

5. Click **Next**. The JMS Destination Properties page appears.

6. In the **Name** field, type a name for the queue, for example, `ORM LoaderQueue`.

7. In the **JNDI Name** field, type the path for accessing the topic as `orm/queue/LoaderQueue`.

8. In the **Template** field, select **None**.

9. Click **Next**.

10. Select **ORM JMSSubdeployment** that you created in the Step 4 of the "Configuring JMS Module Subdeployment" on page A-7 section from the list.

11. Click **Finish**.

To configure the ORM Incoming Event Queue:

1. In the domain tree, select **Services**, **Messaging**, **JMS Modules**.

2. Click **ORM JMSModule**.

3. Click **New** under Summary of Resources.

4. Select **Queue** from the list of JMS resources.

5. Click **Next**. The JMS Destination Properties page appears.

6. In the **Name** field, type a name for the queue, for example, ORM IncomingEventQueue.

7. In the **JNDI Name** field, type the path for accessing the topic as orm/queue/IncomingEventQueue.

8. In the **Template** field, select **None**.

9. Click **Next**.

10. Select **ORM JMSSubdeployment** that you created in the Step 4 of the "Configuring JMS Module Subdeployment" on page A-7 section from the list.

11. Click **Finish**.

A.13 Configuring Data Upload Size Limit

You can upload a DAR file to load data of maximum size 10 MB into the system. If you try to load data larger than this maximum upload size limit, you get an error message. You can configure the maximum data upload size limit to a higher or lower value than the default settings.

To configure the data upload size limit:

1. Go to Environment, Servers, ORM Server.

2. On the Configuration tab, click the **Server Start** subtab.

3. In the **Arguments** field, append the following argument to the new value.

   ```
   -Doracle.iam.rm.loader.max_upload_size=<new value>
   ```

   For example:

   ```
   -Doracle.iam.rm.loader.max_upload_size=1073741824
   ```

4. Click **Save**.

A.14 Creating the Oracle Role Manager User

Before installing the WebLogic server application, you must perform the following steps in the security realm:

1. Go to Security Realms and click the realm name, for example, **myrealm**.

2. Click **Users and Groups** and then **Users**.

3. Click **New** and then type the following details:

    a. In the **Name** field, type `ormserver`.

    b. In the **Description** field, type the description, for example, principal that acts as ORM Server.

    c. In the **Password** field, type the password, for example, `ormserver`.

4. Click **OK**.

5. Click the newly created user, for example, **ormserver**.

6. To add the created user to the Administrators group:

    a. Go to the **Groups** tab.

    b. From the Available list, select **Administrators**.

    c. Click the right-pointing arrow.

    d. Click **Save**.

A.15 Deploying Role Manager

Click **Deployment** to deploy the Role Manager applications and perform the steps described in the "Deploying Role Manager" on page 4-6.

Index