Oracle® Fusion Middleware

Installing and Configuring Oracle Identity and Access Management

12c (12.1.2.4.0)
E95111-12
December 2022

ORACLE

Oracle Fusion Middleware Installing and Configuring Oracle Identity and Access Management, 12c (12.1.2.4.0)

E95111-12

Primary Author: Oracle Corporation

Contents

5 Next Steps After Configuring the Domain

6 Configuring High Availability for Oracle Identity Governance Components

7 Configuring High Availability for Oracle Access Manager Components

Preface

This document describes how to install and configure Oracle Identity and Access Management.

Audience

This guide is intended for system administrators or application developers who are installing and configuring Oracle Identity and Access Management. It is assumed that readers are familiar with web technologies and have a general understanding of Windows and UNIX platforms.

Documentation Accessibility

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

Access to Oracle Support

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

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Related Documents

For more information, see the following documents in the 12c (12.2.1.4.0) documentation set:

- For installation information, see Fusion Middleware Installation Documentation.
- For upgrade information, see Fusion Middleware Upgrade Documentation.
- For administration-related information, see Fusion Middleware Administration Documentation.
- For release-related information, see Fusion Middleware Release Notes.

Conventions

Learn about the conventions used in this document.

This document uses the following text conventions:

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

About the Oracle Identity and Access Management Installation

The standard installation for Oracle Identity and Access Management described in this guide, using which you can install and configure Oracle Access Management and Oracle Identity Governance products.
Oracle Identity and Access Management 12c (12.2.1.4.0) suite has two components: Oracle Access Management (OAM) and Oracle Identity Governance (OIG).

> **Note:**
>
> The product Oracle Identity Manager is referred to as Oracle Identity Manager (OIM) and Oracle Identity Governance (OIG) interchangeably in the guide.

- About Supported Installation Methods
  Oracle Identity and Access Management supports two methods of installation — Simplified and Traditional.
- Using the Standard Installation Topology as a Starting Point
  The standard installation topology is a flexible topology that you can use as a starting point in production environments.

About Supported Installation Methods

Oracle Identity and Access Management supports two methods of installation — Simplified and Traditional.

- **Simplified method:** by using an quick start installer to install all the products in one go.

  > **Note:**
  >
  > Oracle Access Management does not support simplified method.

- **Traditional method:** by individually installing the required products.

The following table shows the supported methods for installing and configuring the components of Oracle Identity and Access Management:

Table 1-1 Methods of Installation for Oracle Identity and Access Management

Component	Simplified Install Method	Traditional Install Method
N/A	**Supported?**	**Supported?**
Oracle Access Management	No	Yes

Table 1-1 (Cont.) Methods of Installation for Oracle Identity and Access Management

Component	Simplified Install Method	Traditional Install Method
Oracle Identity Governance	Yes	Yes

Using the Standard Installation Topology as a Starting Point

The standard installation topology is a flexible topology that you can use as a starting point in production environments.

The information in this guide helps you to create a standard installation topology for Oracle Identity and Access Management. If required, you can later extend the standard installation topology to create a secure and highly available production environment, see Next Steps After Configuring the Domain.

The standard installation topology represents a sample topology for this product. It is not the only topology that this product supports. See About the Standard Installation Topology in *Planning an Installation of Oracle Fusion Middleware*.

- About the Oracle Identity and Access Management Standard Installation Topology
  This topology represents a standard WebLogic Server domain that contains an Administration Server and one or more clusters containing one or more Managed Servers.
- About Elements in the Standard Installation Topology Illustration
  The standard installation topology typically includes common elements.

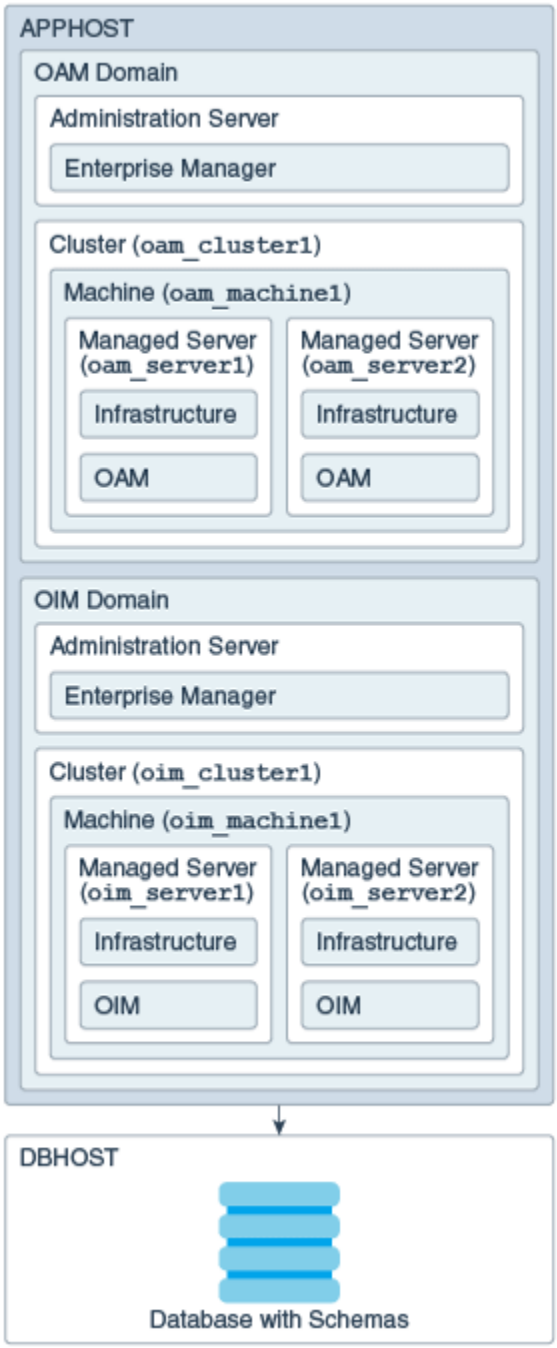
About the Oracle Identity and Access Management Standard Installation Topology

This topology represents a standard WebLogic Server domain that contains an Administration Server and one or more clusters containing one or more Managed Servers.

The following figure shows the standard installation topology for Oracle Identity and Access Management.

See Table 1-2 for information on elements of this topology.

Figure 1-1 Standard Topology for Oracle Identity and Access Management



For Oracle Access Management configuration instructions, see Configuring the Oracle Access Management Domain.

For Oracle Identity Governance configuration instructions, see Configuring the Oracle Identity Governance Domain.

About Elements in the Standard Installation Topology Illustration

The standard installation topology typically includes common elements.

The following table describes all elements of the topology illustration:

Table 1-2 Description of Elements in Standard Installation Topologies

Element	Description and Links to Related Documentation
APPHOST	A standard term used in Oracle documentation to refer to the machine that hosts the application tier.
DBHOST	A standard term used in Oracle documentation to refer to the machine that hosts the database.
WebLogic Domain	A logically related group of Java components (in this case, the Administration Server, Managed Servers, and other related software components). See What Is an Oracle WebLogic Server Domain? in *Understanding Oracle Fusion Middleware*.
Administration Server	Central control entity of a WebLogic domain. It maintains configuration objects for that domain and distributes configuration changes to Managed Servers. See What Is the Administration Server? in *Understanding Oracle Fusion Middleware*.
Enterprise Manager	The Oracle Enterprise Manager Fusion Middleware Control is a primary tool used to manage a domain. See Oracle Enterprise Manager Fusion Middleware Control in *Understanding Oracle Fusion Middleware*.
Cluster	A collection of multiple WebLogic Server instances running simultaneously and working together. See Overview of Managed Servers and Managed Server Clusters in *Understanding Oracle Fusion Middleware*.
Machine	A logical representation of the computer that hosts one or more WebLogic Server instances (servers). Machines are also the logical glue between the Managed Servers and the Node Manager. In order to start or stop the Managed Servers using the Node Manager, associate the Managed Servers with a machine.
Managed Server	A host for your applications, application components, web services, and their associated resources. See Overview of Managed Servers and Managed Server Clusters in *Understanding Oracle Fusion Middleware*.
Infrastructure	A collection of services that include the following: • Metadata repository (MDS) contains the metadata for Oracle Fusion Middleware components, such as the Oracle Application Developer Framework. See What Is the Metadata Repository? in *Understanding Oracle Fusion Middleware*. • Oracle Application Developer Framework (Oracle ADF). • Oracle Web Services Manager (OWSM).

2

Preparing to Install and Configure Oracle Identity and Access Management

To prepare for your Oracle Identity and Access Management installation, verify that your system meets the basic requirements, then obtain the correct installation software.

> **Note:**
>
> The product Oracle Identity Manager is referred to as Oracle Identity Manager (OIM) and Oracle Identity Governance (OIG) interchangeably in the guide.

- Roadmap for Installing and Configuring a Standard Installation Topology
  This roadmap provides the steps required to install and configure Oracle Access Management and Oracle Identity Governance.
- Roadmap for Verifying Your System Environment
  Before you begin the installation and configuration process, you must verify your system environment.
- Obtaining the Product Distribution
  You can obtain the Oracle Fusion Middleware Infrastructure and Oracle Identity and Access Management distribution on *Technical Resources from Oracle*.
- List of Supported Languages

Roadmap for Installing and Configuring a Standard Installation Topology

This roadmap provides the steps required to install and configure Oracle Access Management and Oracle Identity Governance.

Table 2-1 provides the high-level steps required for installing a standard installation topology.

Table 2-1 Standard Installation Roadmap

Task	Description	Documentation
Verify your system environment.	Before you begin the installation, verify that the minimum system and network requirements are met.	See Roadmap for Verifying Your System Environment.
Check for any mandatory patches that are required before the installation.	Review the Oracle Fusion Middleware Infrastructure release notes to see if there are any mandatory patches required for the software products that you are installing.	See Install and Configure in *Release Notes for Oracle Fusion Middleware Infrastructure*. **Note:** For Oracle Identity Governance, if you plan to configure a High Availability setup and have Oracle HTTP Servers (OHS) 12.2.1.4 in your environment with IPV6, then apply patch 31900098.

Table 2-1 (Cont.) Standard Installation Roadmap

Task	Description	Documentation
Obtain the appropriate distributions.	For **Oracle Access Management (OAM)**: Obtain the following distributions: • `fmw_12.2.1.4.0_infrastructure.jar` • `fmw_12.2.1.4.0_idm.jar` --- For **Oracle Identity Governance (OIG):** If you choose to install all the software in one go, obtain the following distributions: • `fmw_12.2.1.4.0_idmquickstart.jar` • `fmw_12.2.1.4.0_idmquickstart2.jar` If you choose to individually install each product, obtain the following distributions: • `fmw_12.2.1.4.0_infrastructure.jar` • `fmw_12.2.1.4.0_soa.jar` • `fmw_12.2.1.4.0_idm.jar`	See Obtaining the Product Distribution. **Note:** After downloading the required `.zip` file, unzip the `.zip` file to obtain the `.jar` distributions. For information about supported installation methods for Oracle Identity and Access Management, see About Supported Installation Methods.
Determine your installation directories.	Verify that the installer can access or create the required installer directories. Also, verify that the directories exist on systems that meet the minimum requirements.	See What Are the Key Oracle Fusion Middleware Directories? in *Understanding Oracle Fusion Middleware*.
Install prerequisite software.	If you are configuring OAM 12.2.1.4.0, you must install Oracle Fusion Middleware Infrastructure 12.2.1.4.0. If you are configuring OIG: • Simplified method: No prerequisites • Traditional method: You must first install Oracle Fusion Middleware Infrastructure 12.2.1.4.0 and then install Oracle SOA Suite 12.2.1.4.0.	See Installing the Infrastructure Software in *Installing and Configuring the Oracle Fusion Middleware Infrastructure*. See Installing the Oracle SOA Suite and Oracle Business Process Management Software in *Installing and Configuring Oracle SOA Suite and Business Process Management*.
Install the software.	Run the Oracle Identity and Access Management installer to install the OAM and OIG binaries. **Note:** If you are using both Oracle Identity Governance and OIG then you must install them in separate *`ORACLE_HOMEs`*. Installing the software transfers the software to your system and creates the Oracle home directory.	For OAM, see Installing the Oracle Access Management Software. For OIG, see Installing the Oracle Identity Governance Software. For an OAM and OIG integrated environment, see Integrating Oracle Identity Governance and Oracle Access Manager Using LDAP Connectors in *Integration Guide for Oracle Identity Management Suite*.

Table 2-1 (Cont.) Standard Installation Roadmap

Task	Description	Documentation
Select a database profile and review any required custom variables.	Before you install the required schemas in the database, review the information about any custom variables you need to set for the Oracle Identity and Access Management schemas.	See About Database Requirements for an Oracle Fusion Middleware Installation.
Create the schemas.	Run the Repository Creation Utility to create the schemas required for configuration.	For OAM, see Creating the Database Schemas. For OIG, see Creating the Database Schemas.
Create a WebLogic domain.	Use the Configuration Wizard/ Assistant to create and configure the WebLogic domain. **Note:** Configure OAM and OIG in two different *DOMAIN_HOMEs*.	For OAM, see Configuring the Oracle Access Management Domain. For OIG, see Configuring the Oracle Identity Governance Domain.
Administer and prepare your domain for high availability.	Discover additional tools and resources to administer your domain and configure your domain to be highly available.	See Next Steps After Configuring the Domain.

Roadmap for Verifying Your System Environment

Before you begin the installation and configuration process, you must verify your system environment.

Table 2-2 identifies important tasks and checks to perform to ensure that your environment is prepared to install and configure Oracle Identity and Access Management.

Table 2-2 Roadmap for Verifying Your System Environment

Task	Description	Documentation
Verify certification and system requirements.	Verify that your operating system is certified and configured for installation and configuration.	See Verifying Certification, System, and Interoperability Requirements.
Identify a proper installation user.	Verify that the installation user has the required permissions to install and configure the software.	See Selecting an Installation User.
Select the installation and configuration directories on your system.	Verify that you can create the necessary directories to install and configure the software, according to the recommended directory structure.	See About the Directories for Installation and Configuration.
Install a certified JDK.	The installation program for the distribution requires a certified JDK present on your system.	See About JDK Requirements for an Oracle Fusion Middleware Installation.

Table 2-2 (Cont.) Roadmap for Verifying Your System Environment

Task	Description	Documentation
Install and configure a database for mid-tier schemas.	To configure your WebLogic domain, you must have access to a certified database that is configured for the schemas required by Oracle Identity and Access Management.	See About Database Requirements for an Oracle Fusion Middleware Installation.

- Verifying Certification, System, and Interoperability Requirements
  Oracle recommends that you use the certification matrix and system requirements documents with each other to verify that your environment meets the requirements for installation.
- Selecting an Installation User
  The user who installs and configures your system must have the required permissions and privileges.
- About the Directories for Installation and Configuration
  During the installation and domain configuration process, you must plan on providing the locations for these directories: Oracle home, Domain home, and the Application home.
- About JDK Requirements for an Oracle Fusion Middleware Installation
  Most Fusion Middleware products are in `.jar` file format. These distributions do not include a JDK. To run a `.jar` distribution installer, you must have a certified JDK installed on your system.
- About Database Requirements for an Oracle Fusion Middleware Installation
  Many Oracle Fusion Middleware products require database schemas prior to configuration. If you do not already have a database where you can install these schemas, you must install and configure a certified database.

Verifying Certification, System, and Interoperability Requirements

Oracle recommends that you use the certification matrix and system requirements documents with each other to verify that your environment meets the requirements for installation.

1. **Verifying that your environment meets certification requirements:**

   Make sure that you install your product on a supported hardware and software configuration. See the certification document for your release on the *Oracle Fusion Middleware Supported System Configurations* page.

   Oracle has tested and verified the performance of your product on all certified systems and environments. Whenever new certifications are released, they are added to the certification document right away. New certifications can be released at any time. Therefore, the certification documents are kept outside the documentation libraries and are available on Oracle Technology Network.

2. **Using the system requirements document to verify certification:**

   Oracle recommends that you use the *Oracle Fusion Middleware System Requirements and Specifications* document to verify that the certification requirements are met. For example, if the certification document indicates that your product is certified for installation on 64-Bit Oracle Linux 6.5, use this

document to verify that your system meets the required minimum specifications. These include disk space, available memory, specific platform packages and patches, and other operating system-specific requirements. System requirements can change in the future. Therefore, the system requirement documents are kept outside of the documentation libraries and are available on Oracle Technology Network.

3. **Verifying interoperability among multiple products:**

   To learn how to install and run multiple Fusion Middleware products from the same release or mixed releases with each other, see Oracle Fusion Middleware Interoperability and Compatibility in *Understanding Interoperability and Compatibility*.

Selecting an Installation User

The user who installs and configures your system must have the required permissions and privileges.

- About User Permissions
  The user who installs a Fusion Middleware product owns the files and has certain permissions on the files.
- About Non-Default User Permissions on UNIX Operating Systems
  Changing the default permission setting reduces the security of the installation and your system. Oracle does not recommend that change the default permission settings.
- Verifying that the Installation User has Administrator Privileges on Windows Operating Systems
  To update the Windows Registry, you must have administrator privileges.

About User Permissions

The user who installs a Fusion Middleware product owns the files and has certain permissions on the files.

- Read and write permissions on all non-executable files (for example, `.jar`, `.properties`, or `.xml`). All other users in the same group as the file owner have read permissions only.
- Read, write, and execute permissions on all executable files (for example, `.exe`, `.sh`, or `.cmd`). All other users in the same group as the file owner have read and execute permissions only.

This means that someone other than the person who installs the software can use the installed binaries in the Oracle home directory to configure a domain or set of Fusion Middleware products.

During configuration, the files generated by the configuration process are owned by the user who ran the Configuration Wizard. This user has the same permissions as described above for the installation user. However, security-sensitive files are not created with group permissions. Only the user that created the domain has read and write permissions and can administer the domain.

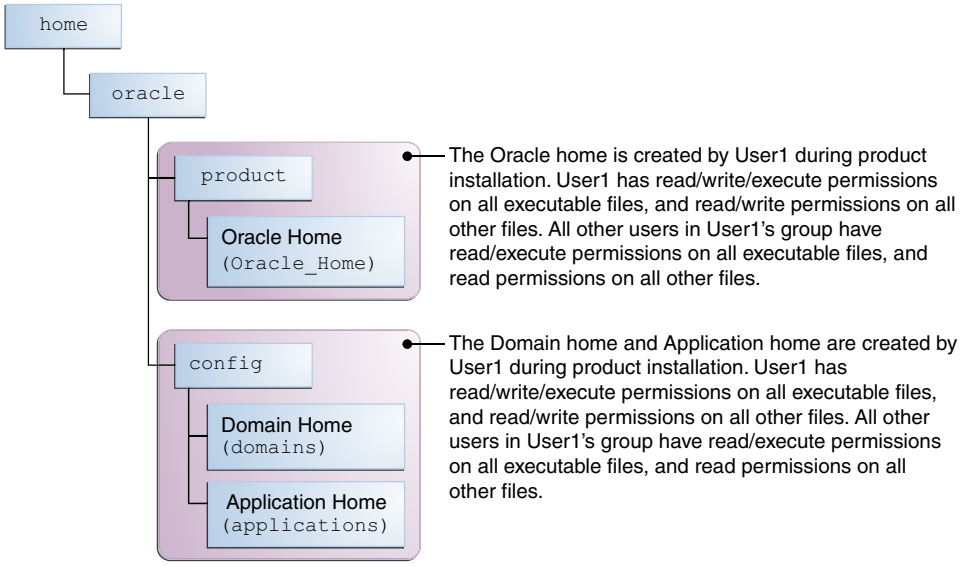
Consider the following examples:

- **Example 1: A Single User Installs the Software and Configures the Domain**

  This example explains the file permissions where the same user installs the software and configures the domain.

To ensure proper permissions and privileges for all files, Oracle recommends that the same owner perform both tasks: install the Oracle Fusion Middleware product and configure the WebLogic Server domain by using the Configuration Wizard.

Figure 2-1 Directory Structure when a Single User Installs the Software and Configures the Domain

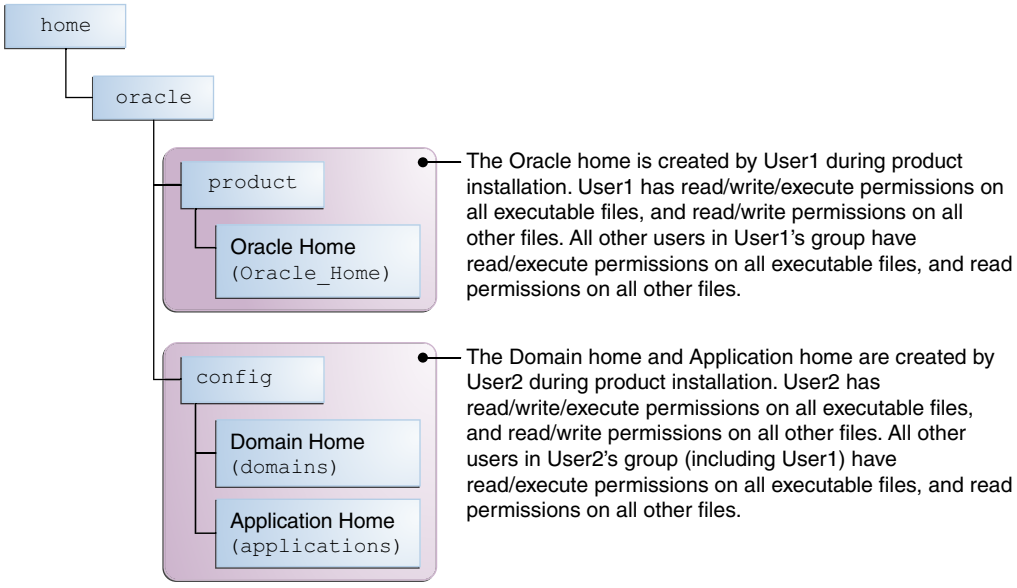


If the user who creates the domain is different than the user who installed the software, then both users must have the same privileges, as shown in the next example.

- **Example 2: The Oracle Home Directory and Domain are Created by Different Users**

  This example explains the file permissions where one user creates the Oracle home and another user configures the domain.

Figure 2-2 Directory Structure when Different Users Install the Software and Configure the Domain



Note:

Certain domain files do not have group permissions. For example, `cwallet.sso`.

Consider the following points before you run the installer:

- On UNIX operating systems, Oracle recommends that you set `umask` to `027` on your system before you install the software. This ensures that the file permissions are set properly during installation. Use the following command:

  ```
  umask 027
  ```

  You must enter this command in the same terminal window from which you plan to run the product installer.
- On UNIX operating systems, do not run the installation program as a `root` user. If you run the installer as a root user, the startup validation may fail and you cannot continue the installation.
- When you manage a product installation (for example, applying patches or starting managed Servers), use the same user ID that you used to install the product.
- On Windows operating systems, you must have administrative privileges to install the product. See Verifying the Installation User has Administrator Privileges on Windows Operating Systems.

About Non-Default User Permissions on UNIX Operating Systems

Changing the default permission setting reduces the security of the installation and your system. Oracle does not recommend that change the default permission settings.

If other users require access to a particular file or executable, use the UNIX `sudo` command or other similar commands to change the file permissions.

Refer to your UNIX operating system Administrator's Guide or contact your operating system vendor, if you need further assistance.

Verifying that the Installation User has Administrator Privileges on Windows Operating Systems

To update the Windows Registry, you must have administrator privileges.

By default, users with the administrator privilege sign in to the system with regular privileges, but can request elevated permissions to perform administrative tasks.

To perform a task with elevated privileges:

1. Find the Command Prompt icon, either from the Start menu or the Windows icon in the lower-left corner.
2. Right-click **Command Prompt** and select **Run as administrator**.

   This opens a new command prompt window, and all actions performed in this window are done with administrator privileges.

   > **Note:**
   >
   > If you have User Access Control enabled on your system, you may see an additional window asking you to confirm this action. Confirm and continue with this procedure.

   > **Note:**
   >
   > For Oracle Identity and Access Management components, ensure that you have enabled User Account Control (UAC). If you have not done already, enable it using the instructions described in Enabling User Account Control (UAC) in the *Oracle Fusion Middleware System Requirements and Specifications*.

3. Perform the desired task.

   For example, to start the product installer:

   For a jar file, enter:

   ```
   java –jar distribution_name.jar
   ```

   For an executable (`.exe`, `.bin`, or `.sh` file), enter:

   ```
   distribution_name.exe
   ```

About the Directories for Installation and Configuration

During the installation and domain configuration process, you must plan on providing the locations for these directories: Oracle home, Domain home, and the Application home.

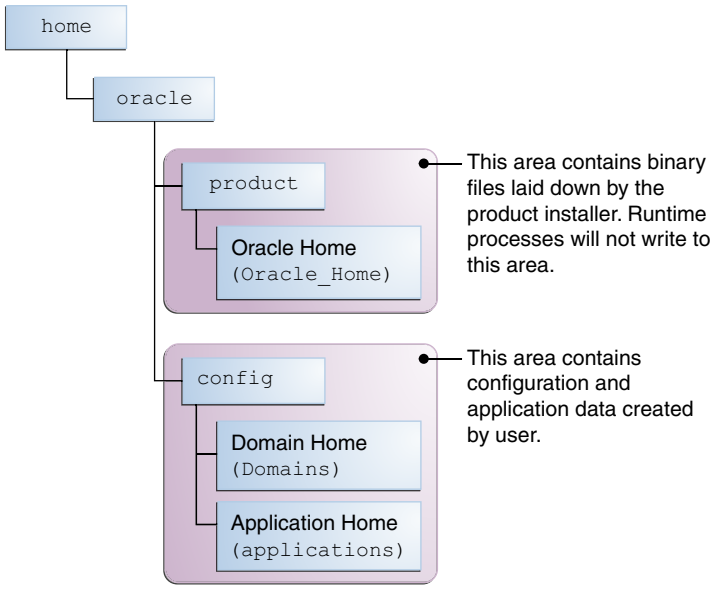
- About the Recommended Directory Structure
  Oracle recommends specific locations for the Oracle Home, Domain Home, and Application Home.
- About the Oracle Home Directory
  When you install any Oracle Fusion Middleware product, you must use an Oracle home directory.
- About the Domain Home Directory
  The Domain home is the directory where domains that you configure are created.
- About the Application Home Directory
  The Application home is the directory where applications for domains you configure are created.
- Preparing for Shared Storage
  Oracle Fusion Middleware allows you to configure multiple WebLogic Server domains from a single Oracle home. This allows you to install the Oracle home in a single location on a shared volume and reuse the Oracle home for multiple host installations.

About the Recommended Directory Structure

Oracle recommends specific locations for the Oracle Home, Domain Home, and Application Home.

Oracle recommends a directory structure similar to the one shown in Figure 2-3.

Figure 2-3 Recommended Oracle Fusion Middleware Directory Structure



A base location (Oracle base) should be established on your system (for example, `/home/oracle`). From this base location, create two separate branches, namely, the `product` directory and the `config` directory. The `product` directory should contain the product binary files and all the Oracle home directories. The `config` directory should contain your domain and application data.

Oracle recommends that you do not keep your configuration data in the Oracle home directory; if you upgrade your product to another major release, are required to create a new Oracle home for binaries. You must also make sure that your configuration data exists in a location where the binaries in the Oracle home have access.

The `/home/oracle/product` (for the Oracle home) and `/home/oracle/config` (for the application and configuration data) directories are used in the examples throughout the documentation; be sure to replace these directories with the actual directories on your system.

About the Oracle Home Directory

When you install any Oracle Fusion Middleware product, you must use an Oracle home directory.

This directory is a repository for common files that are used by multiple Fusion Middleware products installed on the same machine. These files ensure that Fusion Middleware operates correctly on your system. They facilitate checking of cross-product dependencies during installation. For this reason, you can consider the Oracle home directory a *central support directory* for all Oracle Fusion Middleware products installed on your system.

Fusion Middleware documentation refers to the Oracle home directory as *ORACLE_HOME*.

Oracle Home Considerations

Keep the following in mind when you create the Oracle home directory and install Fusion Middleware products:

- Do not include spaces in the name of your Oracle home directory; the installer displays an error message if your Oracle home directory path contains spaces.
- You can install only one instance of each Oracle Fusion Middleware product in a single Oracle home directory. If you need to maintain separate versions of a product on the same machine, each version must be in its own Oracle home directory.

  Although you can have several different products in a single Oracle home, only one version of each product can be in the Oracle home.

Multiple Home Directories

Although in most situations, a single Oracle home directory is sufficient, it is possible to create more than one Oracle home directory. For example, you need to maintain multiple Oracle home directories in the following situations:

- You prefer to maintain separate development and production environments, with a separate product stack for each. With two directories, you can update your development environment without modifying the production environment until you are ready to do so.
- You want to maintain two different versions of a Fusion Middleware product at the same time. For example, you want to install a new version of a product while keeping your existing version intact. In this case, you must install each product version in its own Oracle home directory.
- You need to install multiple products that are not compatible with each other. See Oracle Fusion Middleware 12c (12.2.1.4.0) Interoperability and Compatibility in *Understanding Interoperability and Compatibility* .

> **Note:**
>
> If you create more than one Oracle home directory, you must provide non-overlapping port ranges during the configuration phase for each product.

About the Domain Home Directory

The Domain home is the directory where domains that you configure are created.

The default Domain home location is *ORACLE_HOME*/user_projects/domains/*domain_name*. However, Oracle strongly recommends that you do not use this default location. Put your Domain home *outside* of the Oracle home directory, for example, in /home/oracle/config/domains. The config directory should contain domain and application data. Oracle recommends a separate domain directory so that new installs, patches, and other operations update the *ORACLE_HOME* only, *not* the domain configuration.

> **Note:**
>
> Use different *domain_name*s for Oracle Access Management and Oracle Identity Governance.

See About the Recommended Directory Structure for more on the recommended directory structure and locating your Domain home.

Fusion Middleware documentation refers to the Domain home directory as *DOMAIN_HOME* and includes all folders up to and including the domain name. For example, if you name your domain exampledomain and locate your domain data in the /home/oracle/config/domains directory, the documentation would use *DOMAIN_HOME* to refer to /home/oracle/config/domains/exampledomain.

About the Application Home Directory

The Application home is the directory where applications for domains you configure are created.

The default Application home location is *ORACLE_HOME*/user_projects/applications/*domain_name*. However, Oracle strongly recommends that you locate your Application home *outside* of the Oracle home directory; if you upgrade your product to another major release, you must create a new Oracle home for binaries.

See About the Recommended Directory Structure for more on the recommended directory structure and locating your Application home.

Fusion Middleware documentation refers to the Application home directory as *APPLICATION_HOME* and includes all folders up to and including the domain name. For example, if you name your domain exampledomain and you locate your application data in the /home/oracle/config/applications directory, the documentation uses *APPLICATION_HOME* to refer to /home/oracle/config/applications/exampledomain.

Preparing for Shared Storage

Oracle Fusion Middleware allows you to configure multiple WebLogic Server domains from a single Oracle home. This allows you to install the Oracle home in a single location on a shared volume and reuse the Oracle home for multiple host installations.

If you plan to use shared storage in your environment, see Using Shared Storage in *High Availability Guide* for more information.

About JDK Requirements for an Oracle Fusion Middleware Installation

Most Fusion Middleware products are in `.jar` file format. These distributions do not include a JDK. To run a `.jar` distribution installer, you must have a certified JDK installed on your system.

Make sure that the JDK is installed *outside* of the Oracle home. If you install the JDK under the Oracle home, you may encounter problems when you try to perform tasks in the future. Oracle Universal Installer validates that the Oracle home directory is empty; the install does not progress until you specify an empty directory. Oracle recommends that you locate your JDK installation in the `/home/oracle/products/jdk` directory.

Platform-specific distributions have a `.bin` (for UNIX operating systems) or `.exe` (for Windows operating systems) installer; in these cases, a platform-specific JDK is in the distribution and you do not need to install a JDK separately. However, you may need to upgrade this JDK to a more recent version, depending on the JDK versions that are certified.

Always verify the required JDK version by reviewing the certification information on the *Oracle Fusion Middleware Supported System Configurations* page. For 12c (12.2.1.4.0), the certified JDK is 1.8.0_211 and later.

To download the required JDK, navigate to the following URL and download the Java SE JDK:

http://www.oracle.com/technetwork/java/javase/downloads/index.html

About Database Requirements for an Oracle Fusion Middleware Installation

Many Oracle Fusion Middleware products require database schemas prior to configuration. If you do not already have a database where you can install these schemas, you must install and configure a certified database.

> **Note:**
>
> Multi-tenancy feature is supported, that is, Pluggable Database (PDB) and Container Database (CDB) are supported.

To find a certified database for your operating system, see the certification document for your release on the *Oracle Fusion Middleware Supported System Configurations* page on *Technical Resources from Oracle*.

To make sure that your database is properly configured for schema creation, see Repository Creation Utility Requirements in the *Oracle Fusion Middleware System Requirements and Specifications* document.

For Oracle Identity Governance only:

- For OIG you need to run a few prerequisite `sql` files prior to creating Oracle Identity Management schemas. See Oracle Database requirements when using Oracle Identity Governance in the *System Requirements and Specifications* document.
- Based on your deployment topology and the work load, it is recommended that you refer to the following note on My Oracle Support, and take appropriate actions for your deployment. See Performance Tuning Guidelines and Diagnostics Collection for Oracle Identity Manager (OIM) (Doc ID 1539554.1)

After your database is properly configured, you use the Repository Creation Utility (RCU) to create product schemas in your database. This tool is available in the Oracle home for your Oracle Fusion Middleware product. See About the Repository Creation Utility in *Creating Schemas with the Repository Creation Utility*.

Obtaining the Product Distribution

You can obtain the Oracle Fusion Middleware Infrastructure and Oracle Identity and Access Management distribution on *Technical Resources from Oracle*.

To prepare to install Oracle Fusion Middleware Infrastructure and Oracle Identity and Access Management:

1. Enter `java -version` on the command line to verify that a certified JDK is installed on your system. For 12c (12.2.1.4.0), the certified JDK is 1.8.0_211 and later.

   See About JDK Requirements for an Oracle Fusion Middleware Installation.
2. Locate and download the Oracle Fusion Middleware Infrastructure and Oracle Identity and Access Management software. To configure Oracle Identity Governance in traditional mode, you must download Oracle SOA Suite 12.2.1.4.0.

   See Obtaining Product Distributions in *Planning an Installation of Oracle Fusion Middleware*.

After preparing to install and configure the software:

- For Oracle Access Management, go to Chapter 3: Installing and Configuring the Oracle Access Management Software.
- For Oracle Identity Governance, go to Chapter 4: Installing and Configuring the Oracle Identity Management Software.

List of Supported Languages

Oracle Identity and Access Manager supports the following languages:

Brazilian Portuguese, French, German, Italian, Japanese, Korean, Simplified Chinese, Spanish, Traditional Chinese, Arabic, Czech, Danish, Dutch, Finnish, Greek, Hebrew, Hungarian, Norwegian, Polish, Portuguese, Romanian, Russian, Slovak, Swedish, Thai, and Turkish.

> **Note:**
>
> The following special characters are not allowed in the user login name:
>
> [!@#$%^&*()_-+=[{]}\|;:'",<.>?/~

3

Installing and Configuring the Oracle Access Management Software

Follow the steps in this section to install and configure the Oracle Access Management software.

- Installing the Oracle Access Management Software
- Configuring the Oracle Access Management Domain
  After you have installed Oracle Access Management, you can configure the domain, which you can also extend for high availability.

Installing the Oracle Access Management Software

Follow the steps in this section to install the Oracle Access Management software. Before beginning the installation, ensure that you have verified the prerequisites and completed all steps covered in Preparing to Install and Configure.

The *only* supported method of installation for Oracle Access Management 12c (12.2.1.4.0) is the traditional method, where you individually install Oracle Fusion Middleware Infrastructure and then install Oracle Access Management.

Dependant Software for Oracle Access Management:

- Oracle Fusion Middleware Infrastructure 12c (12.2.1.4.0)

For information about installing Oracle Fusion Middleware Infrastructure 12c (12.2.1.4.0), see Installing the Infrastructure Software in *Installing and Configuring the Oracle Fusion Middleware Infrastructure*.

For information about supported installation methods, see About Supported Installation Methods.

- Verifying the Installation and Configuration Checklist
  The installation and configuration process requires specific information.
- Starting the Installation Program
  Before running the installation program, you must verify the JDK and prerequisite software is installed.
- Navigating the Installation Screens
  The installer shows a series of screens where you verify or enter information.
- Verifying the Installation
  After you complete the installation, verify whether it was successful by completing a series of tasks.

Verifying the Installation and Configuration Checklist

The installation and configuration process requires specific information.

Table 3-1 lists important items that you must know before, or decide during, Oracle Access Management installation and configuration.

Table 3-1 Installation and Configuration Checklist

Information	Example Value	Description
`JAVA_HOME`	`/home/Oracle/Java/jdk1.8.0_211`	Environment variable that points to the Java JDK home directory.
Database host	`examplehost.exampledomain`	Name and domain of the host where the database is running.
Database port	`1521`	Port number that the database listens on. The default Oracle database listen port is `1521`.
Database service name	`orcl.exampledomain`	Oracle databases require a unique service name. The default service name is `orcl`.
DBA username	`SYS`	Name of user with database administration privileges. The default DBA user on Oracle databases is `SYS`.
DBA password	`myDBApw957`	Password of the user with database administration privileges.
`ORACLE_HOME`	`/home/Oracle/`*`product`*`/ORACLE_HOME`	Directory in which you will install your software. This directory will include Oracle Fusion Middleware Infrastructure and Oracle Access Management, as needed.
WebLogic Server hostname	`examplehost.exampledomain`	Host name for Oracle WebLogic Server and Oracle Access Management consoles.
Console port	`7001`	Port for Oracle WebLogic Server and Oracle Access Management consoles.
`DOMAIN_HOME`	`/home/Oracle/config/domains/idm_domain`	Location in which your domain data is stored.
`APPLICATION_HOME`	`/home/Oracle/config/applications/idm_domain`	Location in which your application data is stored.

Table 3-1 (Cont.) Installation and Configuration Checklist

Information	Example Value	Description
Administrator user name for your WebLogic domain	`weblogic`	Name of the user with Oracle WebLogic Server administration privileges. The default administrator user is `weblogic`.
Administrator user password	`myADMpw902`	Password of the user with Oracle WebLogic Server administration privileges.
RCU	*`ORACLE_HOME`*`/oracle_common/bin`	Path to the Repository Creation Utility (RCU).
RCU schema prefix	`oam`	Prefix for names of database schemas used by Oracle Access Management.
RCU schema password	`myRCUpw674`	Password for the database schemas used by Oracle Access Management.
Configuration utility	*`ORACLE_HOME`*`/oracle_common/common/bin`	Path to the Configuration Wizard for domain creation and configuration.

Starting the Installation Program

Before running the installation program, you must verify the JDK and prerequisite software is installed.

To start the installation program:

1. Sign in to the host system.
2. Change to the directory where you downloaded the installation program.
3. You must have installed the Oracle Fusion Middleware Infrastructure 12c (12.2.1.4.0). For instructions, see Installing the Infrastructure Software in *Installing and Configuring the Oracle Fusion Middleware Infrastructure*.
4. Start the installation program by running the `java` executable from the JDK directory. For example:
   - (UNIX) `/home/Oracle/Java/jdk1.8.0_211/bin/java -jar fmw_12.2.1.4.0_idm.jar`
   - (Windows) `C:\home\Oracle\Java\jdk1.8.0_211\bin\java -jar fmw_12.2.1.4.0_idm.jar`

> **Note:**
>
> You can also start the installer in silent mode using a saved response file instead of launching the installer screens. For more about silent or command line installation, see Using the Oracle Universal Installer in Silent Mode in *Installing Software with the Oracle Universal Installer*.

When the installation program appears, you are ready to begin the installation.

Navigating the Installation Screens

The installer shows a series of screens where you verify or enter information.

The following table lists the order in which installer screens appear. If you need additional help with an installation screen, click **Help**.

Table 3-2 Install Screens

Screen	Description
Installation Inventory Setup	On Linux or UNIX operating systems, this screen opens if this is the first time you are installing any Oracle product on this host. Specify the location where you want to create your central inventory. Make sure that the operating system group name selected on this screen has write permissions to the central inventory location. See About the Oracle Central Inventory in *Installing Software with the Oracle Universal Installer*. This screen does not appear on Windows operating systems.
Welcome	Review the information to make sure that you have met all the prerequisites, then click **Next**.
Auto Updates	Select to skip automatic updates, select patches, or search for the latest software updates, including important security updates, through your My Oracle Support account.
Installation Location	Specify your Oracle home directory location. This Oracle home must include Oracle Fusion Middleware Infrastructure 12c (12.2.1.4.0). You can click **View** to verify and ensure that you are installing in the correct Oracle home. **Note:** Ensure that the Oracle Home path does not contain space.
Installation Type	Use the **Collocated** Installation Type. Collocated mode is a type of installation that is managed through WebLogic Server. To install in collocated mode, you must have installed the required dependant softwares.

Table 3-2 (Cont.) Install Screens

Screen	Description
Prerequisite Checks	This screen verifies that your system meets the minimum necessary requirements. To view the list of tasks that gets verified, select **View Successful Tasks**. To view log details, select **View Log**. If any prerequisite check fails, then an error message appears at the bottom of the screen. Fix the error and click **Rerun** to try again. To ignore the error or the warning message and continue with the installation, click **Skip** (not recommended).
Installation Summary	Use this screen to verify installation options you selected. If you want to save these options to a response file, click **Save Response File** and enter the response file location and name. The response file collects and stores all the information that you have entered, and enables you to perform a silent installation (from the command line) at a later time. Click **Install** to begin the installation.
Installation Progress	This screen shows the installation progress. When the progress bar reaches 100% complete, click **Finish** to dismiss the installer, or click **Next** to see a summary.
Installation Complete	This screen displays the Installation Location and the Feature Sets that are installed. Review this information and click **Finish** to close the installer.

Verifying the Installation

After you complete the installation, verify whether it was successful by completing a series of tasks.

- Reviewing the Installation Log Files
  Review the contents of the installation log files to make sure that the installer did not encounter any problems.
- Checking the Directory Structure
  The contents of your installation vary based on the options that you selected during the installation.
- Viewing the Contents of the Oracle Home
  You can view the contents of the Oracle home directory by using the `viewInventory` script.

Reviewing the Installation Log Files

Review the contents of the installation log files to make sure that the installer did not encounter any problems.

By default, the installer writes logs files to the *`Oracle_Inventory_Location`*`/logs` (on UNIX operating systems) or *`Oracle_Inventory_Location`*`\logs` (on Windows operating systems) directory.

For a description of the log files and where to find them, see Installation Log Files in *Installing Software with the Oracle Universal Installer*.

Checking the Directory Structure

The contents of your installation vary based on the options that you selected during the installation.

See What Are the Key Oracle Fusion Middleware Directories? in *Understanding Oracle Fusion Middleware*.

Viewing the Contents of the Oracle Home

You can view the contents of the Oracle home directory by using the `viewInventory` script.

See Viewing the Contents of an Oracle Home in *Installing Software with the Oracle Universal Installer*.

Configuring the Oracle Access Management Domain

After you have installed Oracle Access Management, you can configure the domain, which you can also extend for high availability.

The configuration steps presented here assume that you have completed the installation steps covered in:

- Preparing to Install and Configure
- Installing the Oracle Access Management Software

Refer to the following sections to create the database schemas, configure a WebLogic domain, and verify the configuration:

- Creating the Database Schemas
  Before you can configure a domain, you must install required schemas on a certified database for use with this release of Oracle Fusion Middleware.
- Configuring the Domain
  Use the Configuration Wizard to create and configure a domain.
- Starting the Servers
  After a successful configuration, start all processes and servers, including the Administration Server and any Managed Servers.
- Verifying the Configuration
  After completing all configuration steps, you can perform additional steps to verify that your domain is properly configured.
- Setting the Memory Parameters for OAM Domain (Optional)
  If the initial startup parameter in Oracle Access Management domain, which defines the memory usage, is insufficient, you can increase the value of this parameter.
- Updating the java.security File (Optional)
  If you wish to integrate Oracle Access Management 12c (12.2.1.4.0) with Oracle Adaptive Access Manager (OAAM) 11*g* Release 2 (11.1.2.3.0), you must update `java.security` file with the following changes, post upgrade:

- Troubleshooting
  This section lists the common issues encountered while configuring Oracle Access Management and their workarounds.

Creating the Database Schemas

Before you can configure a domain, you must install required schemas on a certified database for use with this release of Oracle Fusion Middleware.

- Installing and Configuring a Certified Database
  Before you create the database schemas, you must install and configure a certified database, and verify that the database is up and running.
- Starting the Repository Creation Utility
  Start the Repository Creation Utility (RCU) after you verify that a certified JDK is installed on your system.
- Navigating the Repository Creation Utility Screens to Create Schemas
  Enter required information in the RCU screens to create the database schemas.

Installing and Configuring a Certified Database

Before you create the database schemas, you must install and configure a certified database, and verify that the database is up and running.

> **Note:**
>
> For an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), you must modify the wallet settings and set the environment variables as described in Settings to connect to Autonomous Transaction Processing Database, and apply patches on `ORACLE_HOME` as described in Applying Patches on ORACLE HOME.

See About Database Requirements for an Oracle Fusion Middleware Installation.

Starting the Repository Creation Utility

Start the Repository Creation Utility (RCU) after you verify that a certified JDK is installed on your system.

To start the RCU:

1. Verify that a certified JDK already exists on your system by running `java -version` from the command line. For 12c (12.2.1.4.0), the certified JDK is 1.8.0_211 and later.

   See About JDK Requirements for an Oracle Fusion Middleware Installation.
2. Ensure that the `JAVA_HOME` environment variable is set to the location of the certified JDK. For example:
   - (UNIX) `setenv JAVA_HOME /home/Oracle/Java/jdk1.8.0_211`
   - (Windows) `set JAVA_HOME=C:\home\Oracle\Java\jdk1.8.0_211`
3. Change to the following directory:

- (UNIX) *ORACLE_HOME*/oracle_common/bin
- (Windows) *ORACLE_HOME*\oracle_common\bin

4. Enter the following command:
   - (UNIX) ./rcu
   - (Windows) rcu.bat

Navigating the Repository Creation Utility Screens to Create Schemas

Enter required information in the RCU screens to create the database schemas.

- Introducing the RCU
  The Welcome screen is the first screen that appears when you start the RCU.
- Selecting a Method of Schema Creation
  Use the Create Repository screen to select a method to create and load component schemas into the database.
- Providing Database Connection Details
  On the Database Connection Details screen, provide the database connection details for the RCU to connect to your database.
- Specifying a Custom Prefix and Selecting Schemas
- Specifying Schema Passwords
  On the Schema Passwords screen, specify how you want to set the schema passwords on your database, then enter and confirm your passwords.
- Completing Schema Creation
  Navigate through the remaining RCU screens to complete schema creation.

Introducing the RCU

The Welcome screen is the first screen that appears when you start the RCU.

Click **Next**.

Selecting a Method of Schema Creation

Use the Create Repository screen to select a method to create and load component schemas into the database.

On the Create Repository screen:

- If you have the necessary permissions and privileges to perform DBA activities on your database, select **System Load and Product Load**. This procedure assumes that you have SYSDBA privileges.
- If you do *not* have the necessary permissions or privileges to perform DBA activities in the database, you must select **Prepare Scripts for System Load** on this screen. This option generates a SQL script that you can give to your database administrator. See About System Load and Product Load in *Creating Schemas with the Repository Creation Utility*.
- If the DBA has already run the SQL script for System Load, select **Perform Product Load**.

> **Note:**
>
> For an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), you must create schemas as a `Normal` user, and though, you do not have full SYS or SYSDBA privileges on the database, you must select **System Load and Product Load**.

Providing Database Connection Details

On the Database Connection Details screen, provide the database connection details for the RCU to connect to your database.

> **Note:**
>
> If you are unsure of the database service name, you can obtain it from the `SERVICE_NAMES` parameter in the initialization parameter file of the database. If the initialization parameter file does not contain the `SERVICE_NAMES` parameter, then the service name is the same as the global database name, which is specified in the `DB_NAME` and `DB_DOMAIN` parameters.
>
> For an Oracle Autonomous Transaction Processing-Shared (ATP-S) database, use the database service name, `<databasename>_tpurgent` or `<databasename>_tp`, specified in `tnsnames.ora`. For service name details, see Database Service Names for Autonomous Transaction Processing and Autonomous JSON Database in *Using Oracle Autonomous Database on Shared Exadata Infrastructure*.

To create schemas on an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), you can specify the connection credentials using only the **Connection String** option. In this screen, a warning message is displayed. You can ignore the warning and continue with the schema creation. For more information, see SYS DBA Privileges Warning After Applying Patches.

To provide the database connection details:

1. On the Database Connection Details screen, provide the database connection details. For example:

   Database Type: `Oracle Database`
   Connection String Format: `Connection Parameters or Connection String`
   Connection String:
   `examplehost.exampledomain.com:1521:Orcl.exampledomain.com`
   Host Name: `examplehost.exampledomain.com`
   Port: `1521`
   Service Name: `Orcl.exampledomain.com`
   User Name: `sys`
   Password: `******`
   Role: `SYSDBA`

For an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), enter connect string in the following format:

```
jdbc:oracle:thin:@TNS_alias?TNS_ADMIN=<path of the wallet files,
ojdbc.properties, and tnsnames.ora>
```

In the connect string, you must pass `TNS_alias` as the database name found in `tnsnames.ora`, and `TNS_ADMIN` property to the location of the wallet files, `ojdbc.properties`, and `tnsnames.ora`.

> **Note:**
>
> For an Oracle Autonomous Transaction Processing-Shared (ATP-S) database, you must use only one of the database service names, `<databasename>_tpurgent` or `<databasename>_tp`, specified in `tnsnames.ora`. For database service name details, see Database Service Names for Autonomous Transaction Processing and Autonomous JSON Database in *Using Oracle Autonomous Database on Shared Exadata Infrastructure*.

Example connect string for Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database::

```
jdbc:oracle:thin:@dbname_medium?TNS_ADMIN=/users/test/wallet_dbname/
```

Example connect string for Oracle Autonomous Transaction Processing-Shared (ATP-S) database:

```
jdbc:oracle:thin:@dbname_tp?TNS_ADMIN=/users/test/wallet_dbname/
```

2. Click **Next** to proceed, then click **OK** in the dialog window that confirms a successful database connection.

Specifying a Custom Prefix and Selecting Schemas

Select **Create new prefix**, specify a custom prefix, then expand **IDM Schemas** and select the **Oracle Access Manager** schema. This action automatically selects the following schemas as dependencies:

- Common Infrastructure Services (STB)
- Oracle Platform Security Services (OPSS)
- Audit Services (IAU)
- Audit Services Append (IAU_Append)
- Audit Services Viewer (IAU_Viewer)
- Metadata Services (MDS)
- WebLogic Services (WLS)

The schema Common Infrastructure Services (STB) is automatically created. This schema is dimmed; you cannot select or deselect it. This schema enables you to retrieve information from RCU during domain configuration. For more information, see "Understanding the Service Table Schema" in *Creating Schemas with the Repository Creation Utility*.

The custom prefix is used to logically group these schemas together for use in this domain only; you must create a unique set of schemas for each domain. Schema sharing across domains is not supported.

> **Tip:**
>
> For more information about custom prefixes, see "Understanding Custom Prefixes" in *Creating Schemas with the Repository Creation Utility*.
>
> For more information about how to organize your schemas in a multi-domain environment, see "Planning Your Schema Creation" in *Creating Schemas with the Repository Creation Utility*.

> **Tip:**
>
> You must make a note of the custom prefix you choose to enter here; you will need this later on during the domain creation process.

Click **Next** to proceed, then click **OK** on the dialog window confirming that prerequisite checking for schema creation was successful.

Specifying Schema Passwords

On the Schema Passwords screen, specify how you want to set the schema passwords on your database, then enter and confirm your passwords.

> **Note:**
>
> For an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), the schema password must be minimum 12 characters, and must contain at least one uppercase, one lower case, and one number.

You must make a note of the passwords you set on this screen; you will need them later on during the domain creation process.

Click **Next**.

Completing Schema Creation

Navigate through the remaining RCU screens to complete schema creation.

On the Map Tablespaces screen, the Encrypt Tablespace check box appears *only* if you enabled Transparent Data Encryption (TDE) in the database (Oracle or Oracle EBR) when you start the RCU.

To complete schema creation:

1. On the Map Tablespaces screen, select **Encrypt Tablespace** if you want to encrypt all new tablespaces that the RCU creates.
2. In the Completion Summary screen, click **Close** to dismiss the RCU.

   For an Oracle Autonomous Transaction Processing-Shared (ATP-S) database, in the **Map Tablespaces** screen you must override the default tablespaces and the temporary tablespaces, and also override the additional tablespaces, if applicable. See Map Tablespaces.

   If you encounter any issues when you create schemas on an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), see Troubleshooting Tips for Schema Creation on an Autonomous Transaction Processing Database in *Creating Schemas with the Repository Creation Utility* and Issues Related to Product Installation and Configuration on an Autonomous Database in *Release Notes for Oracle Fusion Middleware Infrastructure*.

Configuring the Domain

Use the Configuration Wizard to create and configure a domain.

For information on other methods to create domains, see Additional Tools for Creating, Extending, and Managing WebLogic Domains in *Creating WebLogic Domains Using the Configuration Wizard*.

- Starting the Configuration Wizard
  Start the Configuration Wizard to begin configuring a domain.
- Navigating the Configuration Wizard Screens to Create and Configure the Domain
  Enter required information in the Configuration Wizard screens to create and configure the domain for the topology.
- Updating the System Properties for SSL Enabled Servers
  For SSL enabled servers, you must set the required properties in the `setDomainEnv` file in the domain home.

Starting the Configuration Wizard

Start the Configuration Wizard to begin configuring a domain.

To start the Configuration Wizard:

1. Change to the following directory:

   (UNIX) *ORACLE_HOME*`/oracle_common/common/bin`

   (Windows) *ORACLE_HOME*`\oracle_common\common\bin`

   where *ORACLE_HOME* is your 12c (12.2.1.4.0) Oracle home.
2. Enter the following command:

   (UNIX) `./config.sh`

   (Windows) `config.cmd`

Navigating the Configuration Wizard Screens to Create and Configure the Domain

Enter required information in the Configuration Wizard screens to create and configure the domain for the topology.

> **Note:**
>
> You can use this procedure to extend an existing domain. If your needs do not match the instructions in the procedure, be sure to make your selections accordingly, or see the supporting documentation for more details.

- Selecting the Domain Type and Domain Home Location
  Use the Configuration Type screen to select a Domain home directory location, optimally outside the Oracle home directory.
- Selecting the Configuration Templates for Oracle Access Management
- Selecting the Application Home Location
  Use the Application Location screen to select the location to store applications associated with your domain, also known as the *Application home* directory.
- Configuring the Administrator Account
  Use the Administrator Account screen to specify the user name and password for the default WebLogic Administrator account for the domain.
- Specifying the Domain Mode and JDK
  Use the Domain Mode and JDK screen to specify the domain mode and Java Development Kit (JDK).
- Specifying the Database Configuration Type
  Use the Database Configuration type screen to specify details about the database and database schema.
- Specifying JDBC Component Schema Information
  Use the JDBC Component Schema screen to verify or specify details about the database schemas.
- Testing the JDBC Connections
  Use the JDBC Component Schema Test screen to test the data source connections.
- Selecting Advanced Configuration
  Use the Advanced Configuration screen to complete the domain configuration.
- Configuring the Administration Server Listen Address
  Use the Administration Server screen to select the IP address of the host.
- Configuring Node Manager
  Use the Node Manager screen to select the type of Node Manager you want to configure, along with the Node Manager credentials.
- Configuring Managed Servers for Oracle Access Management
- Configuring a Cluster for Oracle Access Management
  Use the Clusters screen to create a new cluster.
- Defining Server Templates
  If you are creating dynamic clusters for a high availability setup, use the Server Templates screen to define one or more server templates for domain.

- Configuring Dynamic Servers
  You can skip this screen for Oracle Access Management configuration.
- Assigning Oracle Access Management Managed Servers to the Cluster
  If you are configuring a non-clustered setup, click **Next** and go to next screen. Use the Assign Servers to Clusters screen to assign Managed Servers to a new *configured cluster*. A configured cluster is a cluster you configure manually. You do not use this screen if you are configuring a *dynamic cluster*, a cluster that contains one or more generated server instances that are based on a server template.
- Configuring Coherence Clusters
  Use the Coherence Clusters screen to configure the Coherence cluster.
- Creating a New Oracle Access Management Machine
  Use the Machines screen to create new machines in the domain. A machine is required so that Node Manager can start and stop servers.
- Assigning Servers to Oracle Access Management Machines
  Use the Assign Servers to Machines screen to assign the Administration Server and Managed Servers to the new machine you just created.
- Virtual Targets
  You can skip this screen for Oracle Access Management configuration.
- Partitions
  The Partitions screen is used to configure partitions for virtual targets in WebLogic Server Multitenant (MT) environments. Select **Next** without selecting any options.
- Configuring Domain Frontend Host
  The Domain Frontend Host screen can be used to configure the frontend host for the domain.
- Targeting the Deployments
  The Deployments Targeting screen can be used to target the available deployments to the servers.
- Targeting the Services
  The Services Targeting screen can be used to target the available services to the Servers.
- Reviewing Your Configuration Specifications and Configuring the Domain
  The Configuration Summary screen shows detailed configuration information for the domain you are about to create.
- Writing Down Your Domain Home and Administration Server URL
  The End of Configuration screen shows information about the domain you just configured.

Selecting the Domain Type and Domain Home Location

Use the Configuration Type screen to select a Domain home directory location, optimally outside the Oracle home directory.

Oracle recommends that you locate your Domain home in accordance with the directory structure in What Are the Key Oracle Fusion Middleware Directories? in *Understanding Oracle Fusion Middleware*, where the Domain home is located outside the Oracle home directory. This directory structure helps avoid issues when you need to upgrade or reinstall software.

To specify the Domain type and Domain home directory:

1. On the Configuration Type screen, select **Create a new domain**.

2. In the Domain Location field, specify your Domain home directory.

For more details about this screen, see Configuration Type in *Creating WebLogic Domains Using the Configuration Wizard*.

Selecting the Configuration Templates for Oracle Access Management

On the Templates screen, make sure **Create Domain Using Product Templates** is selected, then select the template **Oracle Access Management Suite**.

Selecting this template automatically selects the following as dependencies:

- Oracle Enterprise Manager
- Oracle JRF
- WebLogic Coherence Cluster Extension

> **Note:**
>
> The basic WebLogic domain is pre-selected.

More information about the options on this screen can be found in Templates in *Creating WebLogic Domains Using the Configuration Wizard*.

Selecting the Application Home Location

Use the Application Location screen to select the location to store applications associated with your domain, also known as the *Application home* directory.

Oracle recommends that you locate your Application home in accordance with the directory structure in What Are the Key Oracle Fusion Middleware Directories? in *Understanding Oracle Fusion Middleware*, where the Application home is located outside the Oracle home directory. This directory structure helps avoid issues when you need to upgrade or re-install your software.

For more about the Application home directory, see About the Application Home Directory.

For more information about this screen, see Application Location in *Creating WebLogic Domains Using the Configuration Wizard*.

Configuring the Administrator Account

Use the Administrator Account screen to specify the user name and password for the default WebLogic Administrator account for the domain.

Oracle recommends that you make a note of the user name and password that you enter on this screen; you need these credentials later to boot and connect to the domain's Administration Server.

For more information about this screen, see Administrator Account in *Creating WebLogic Domains Using the Configuration Wizard.*

Specifying the Domain Mode and JDK

Use the Domain Mode and JDK screen to specify the domain mode and Java Development Kit (JDK).

On the Domain Mode and JDK screen:

- Select **Production** in the **Domain Mode** field.
- Select the **Oracle HotSpot JDK** in the **JDK** field.

For more information about this screen, see Domain Mode and JDK in *Creating WebLogic Domains Using the Configuration Wizard*.

Specifying the Database Configuration Type

Use the Database Configuration type screen to specify details about the database and database schema.

On the Database Configuration type screen, select **RCU Data**. This option instructs the Configuration Wizard to connect to the database and Service Table (STB) schema to automatically retrieve schema information for schemas needed to configure the domain.

> **Note:**
>
> If you select **Manual Configuration** on this screen, you must manually fill in parameters for your schema on the next screen.
>
> For an Autonomous Transaction Processing database, (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), you must select only the **RCU Data** option.

After selecting **RCU Data**, specify details in the following fields:

Field	Description
DBMS/Service	Enter the database DBMS name, or service name if you selected a service type driver. Example: `orcl.exampledomain.com`
Host Name	Enter the name of the server hosting the database. Example: `examplehost.exampledomain.com`
Port	Enter the port number on which the database listens. Example: `1521`
Schema Owner Schema Password	Enter the username and password for connecting to the database's Service Table schema. This is the schema username and password entered for the Service Table component on the Schema Passwords screen in the RCU (see Specifying Schema Passwords). The default username is *prefix*`_STB`, where *prefix* is the custom prefix that you defined in the RCU.

For an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), specify the connection credentials using only the **Connection URL String** option, and enter the connect string in the following format:

```
jdbc:oracle:thin:@TNS_alias?TNS_ADMIN=<path of the wallet files,
ojdbc.properties, and tnsnames.ora>
```

In the connect string, you must pass `TNS_alias` as the database name found in `tnsnames.ora`, and `TNS_ADMIN` property to the location of the wallet files, `ojdbc.properties`, and `tnsnames.ora`.

Example connect string for Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database:

```
jdbc:oracle:thin:@dbname_medium?TNS_ADMIN=/users/test/wallet_dbname/
```

Example connect string for Oracle Autonomous Transaction Processing-Shared (ATP-S) database:

```
jdbc:oracle:thin:@dbname_tp?TNS_ADMIN=/users/test/wallet_dbname/
```

Click **Get RCU Configuration** when you finish specifying the database connection information. The following output in the Connection Result Log indicates that the operation succeeded:

```
Connecting to the database server...OK
Retrieving schema data from database server...OK
Binding local schema components with retrieved data...OK

Successfully Done.
```

For more information about the schema installed when the RCU is run, see About the Service Table Schema in *Creating Schemas with the Repository Creation Utility*.

See Database Configuration Type in *Creating WebLogic Domains Using the Configuration Wizard* .

Specifying JDBC Component Schema Information

Use the JDBC Component Schema screen to verify or specify details about the database schemas.

Verify that the values populated on the JDBC Component Schema screen are correct for all schemas. If you selected **RCU Data** on the previous screen, the schema table should already be populated appropriately. If you selected **Manual configuration** on the Database Configuration screen, you must configure the schemas listed in the table manually, before you proceed.

For an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), specify the connection credentials using only the **Connection URL String** option, and enter the connect string in the following format:

```
jdbc:oracle:thin:@TNS_alias?TNS_ADMIN=<path of the wallet files,
ojdbc.properties, and tnsnames.ora>
```

In the connect string, you must pass `TNS_alias` as the database service name found in `tnsnames.ora`, and `TNS_ADMIN` property to the location of the wallet files, `ojdbc.properties`, and `tnsnames.ora`.

Example connect string for Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database:

```
jdbc:oracle:thin:@dbname_medium?TNS_ADMIN=/users/test/wallet_dbname/
```

Example connect string for Oracle Autonomous Transaction Processing-Shared (ATP-S) database:

```
jdbc:oracle:thin:@dbname_tp?TNS_ADMIN=/users/test/wallet_dbname/
```

For high availability environments, see the following sections in *High Availability Guide* for additional information on configuring data sources for Oracle RAC databases:

- Configuring Active GridLink Data Sources with Oracle RAC
- Configuring Multi Data Sources

See JDBC Component Schema in *Creating WebLogic Domains Using the Configuration Wizard* for more details about this screen.

Testing the JDBC Connections

Use the JDBC Component Schema Test screen to test the data source connections.

A green check mark in the Status column indicates a successful test. If you encounter any issues, see the error message in the Connection Result Log section of the screen, fix the problem, then try to test the connection again.

By default, the schema password for each schema component is the password you specified while creating your schemas. If you want different passwords for different schema components, manually edit them in the previous screen (JDBC Component Schema) by entering the password you want in the **Schema Password** column, against each row. After specifying the passwords, select the check box corresponding to the schemas that you changed the password in and test the connection again.

For more information about this screen, see JDBC Component Schema Test in *Creating WebLogic Domains Using the Configuration Wizard*.

Selecting Advanced Configuration

Use the Advanced Configuration screen to complete the domain configuration.

On the Advanced Configuration screen, select:

- Administration Server

  Required to properly configure the listen address of the Administration Server.
- Node Manager

  Required to configure Node Manager.
- Topology

  Required to configure the Oracle Access Management Managed Server.

Optionally, select other available options as required for your desired installation environment. The steps in this guide describe a standard installation topology, but you

may choose to follow a different path. If your installation requirements extend to additional options outside the scope of this guide, you may be presented with additional screens to configure those options. For information about all Configuration Wizard screens, see Configuration Wizard Screens in *Creating WebLogic Domains Using the Configuration Wizard*.

Configuring the Administration Server Listen Address

Use the Administration Server screen to select the IP address of the host.

Select the drop-down list next to **Listen Address** and select the IP address of the host where the Administration Server will reside, or use the system name or DNS name that maps to a single IP address. Do *not* use `All Local Addresses`.

Do *not* specify any server groups for the Administration Server.

Configuring Node Manager

Use the Node Manager screen to select the type of Node Manager you want to configure, along with the Node Manager credentials.

Select **Per Domain Default Location** as the Node Manager type, then specify Node Manager credentials.

For more information about this screen, see Node Manager in *Creating WebLogic Domains Using the Configuration Wizard*.

For more about Node Manager types, see Node Manager Overview in *Administering Node Manager for Oracle WebLogic Server*.

Configuring Managed Servers for Oracle Access Management

On the Managed Servers screen, the new Managed Servers named `oam_server_1` and `oam_policy_mgr1` are displayed:

1. In the Listen Address drop-down list, select the IP address of the host on which the Managed Server will reside or use the system name or DNS name that maps to a single IP address. Do not use "All Local Addresses."
2. In the Server Groups drop-down list, select the server group for your managed server. By default, **OAM-MGD-SVRS** is selected for oam_server1 and **OAM-POLICY-MANAGED-SERVER** is selected for oam_policy_mgr1.

   Server groups target Fusion Middleware applications and services to one or more servers by mapping defined application service groups to each defined server group. A given application service group may be mapped to multiple server groups if needed. Any application services that are mapped to a given server group are automatically targeted to all servers that are assigned to that group. For more information, see "Application Service Groups, Server Groups, and Application Service Mappings" in *Domain Template Reference*.
3. Configuring a second Managed Server is one of the steps needed to configure the standard topology for high availability. If you are not creating a highly available environment, then this step is optional.

   Click **Clone** and repeat this process to create a second Managed Server named `oam_policy_mgr2`.

> **Note:**
>
> If you wish to configure additional Managed Servers, use the **Clone** option and add the Managed Server. For example, if we want to configure `oam_server2`, click **Clone** and select **oam_server1** to clone this server. Do not use the **add** option to add a new Managed Server.

Configuring a second Managed Server is one of the steps needed to configure the standard topology for high availability. If you are not creating a highly available environment, then this step is optional.

For more information about the high availability standard topology, see "Understanding the Fusion Middleware Standard HA Topology" in *High Availability Guide*.

For more information about the next steps to prepare for high availability after your domain is configured, see Preparing Your Environment for High Availability.

These server names and will be referenced throughout this document; if you choose different names be sure to replace them as needed.

> **Tip:**
>
> More information about the options on this screen can be found in Managed Servers in *Creating WebLogic Domains Using the Configuration Wizard*.

Configuring a Cluster for Oracle Access Management

Use the Clusters screen to create a new cluster.

> **Note:**
>
> If you are configuring a non-clustered setup on a single node, skip this screen.

On the Clusters screen:

1. Click **Add**.
2. Specify `oam_cluster_1` in the Cluster Name field for `oam_server`. For `oam_policy_mgr` server, you must create another cluster, for example, `oam_policy_cluster`.
3. For the Cluster Address field, specify the *`ipaddress/hostname`*`:port`. For example:

   `ip_address_machine1``:portnumber,`*`ip_address_machine2`*`:portnumber`

By default, server instances in a cluster communicate with one another using unicast. If you want to change your cluster communications to use multicast, see Considerations for Choosing Unicast or Multicast in *Administering Clusters for Oracle WebLogic Server*.

You can also create clusters using Fusion Middleware Control. In this case, you can configure cluster communication (unicast or multicast) when you create the new cluster. See Create and configure clusters in *Oracle WebLogic Server Administration Console Online Help*.

For more information about this screen, see Clusters in *Creating WebLogic Domains Using the Configuration Wizard*.

Defining Server Templates

If you are creating dynamic clusters for a high availability setup, use the Server Templates screen to define one or more server templates for domain.

To continue configuring the domain, click **Next**.

For steps to create a dynamic cluster for a high availability setup, see Using Dynamic Clusters in *High Availability Guide*.

Configuring Dynamic Servers

You can skip this screen for Oracle Access Management configuration.

Click **Next** and proceed.

Assigning Oracle Access Management Managed Servers to the Cluster

If you are configuring a non-clustered setup, click **Next** and go to next screen. Use the Assign Servers to Clusters screen to assign Managed Servers to a new *configured cluster*. A configured cluster is a cluster you configure manually. You do not use this screen if you are configuring a *dynamic cluster*, a cluster that contains one or more generated server instances that are based on a server template.

For more on configured cluster and dynamic cluster terms, see About Dynamic Clusters in *Understanding Oracle WebLogic Server*.

On the Assign Servers to Clusters screen:

1. In the Clusters pane, select the cluster to which you want to assign the Managed Servers; in this case, `oam_cluster_1`.
2. In the Servers pane, assign `oam_server_1` to `oam_cluster_1` by doing one of the following:
   - Click once on `oam_server_1` to select it, then click the right arrow to move it beneath the selected cluster (`oam_cluster_1`) in the Clusters pane.
   - Double-click on `oam_server_1` to move it beneath the selected cluster (`oam_cluster_1`) in the Clusters pane.
3. Repeat to assign `oam_policy_mgr` to `oam_policy_cluster`.

For more information about this screen, see Assign Servers to Clusters in *Creating WebLogic Domains Using the Configuration Wizard*.

Configuring Coherence Clusters

Use the Coherence Clusters screen to configure the Coherence cluster.

Leave the default port number as the Coherence cluster listen port. After configuration, the Coherence cluster is automatically added to the domain.

> **Note:**
>
> Setting the unicast listen port to `0` creates an offset for the Managed Server port numbers. The offset is `5000`, meaning the maximum allowed value that you can assign to a Managed Server port number is `60535`, instead of `65535`.

See Table 5-2 for more information and next steps for configuring Coherence.

For Coherence licensing information, see Oracle Coherence Products in *Licensing Information*.

Creating a New Oracle Access Management Machine

Use the Machines screen to create new machines in the domain. A machine is required so that Node Manager can start and stop servers.

If you plan to create a high availability environment and know the list of machines your target topology requires, you can follow the instructions in this section to create all the machines at this time. For more about scale out steps, see Optional Scale Out Procedure in *High Availability Guide*.

To create a new Oracle Access Management machine so that Node Manager can start and stop servers:

1. Select the Machine tab (for Windows) or the UNIX Machine tab (for UNIX), then click **Add** to create a new machine.
2. In the Name field, specify a machine name, such as `oam_machine_1`.
3. In the Node Manager Listen Address field, select the IP address of the machine in which the Managed Servers are being configured.

   You must select a specific interface and not `localhost`. This allows Coherence cluster addresses to be dynamically calculated.
4. Verify the port in the Node Manager Listen Port field.
5. Repeat these steps to add more machines, if required.

> **Note:**
>
> If you are extending an existing domain, you can assign servers to any existing machine. It is not necessary to create a new machine unless your situation requires it.

For more information about this screen, see Machines in *Creating WebLogic Domains Using the Configuration Wizard*.

Assigning Servers to Oracle Access Management Machines

Use the Assign Servers to Machines screen to assign the Administration Server and Managed Servers to the new machine you just created.

On the Assign Servers to Machines screen:

1. In the Machines pane, select the machine to which you want to assign the servers; in this case, `oam_machine_1`.
2. In the Servers pane, assign `AdminServer` to `oam_machine_1` by doing one of the following:
   - Click once on `AdminServer` to select it, then click the right arrow to move it beneath the selected machine (`oam_machine_1`) in the Machines pane.
   - Double-click on `AdminServer` to move it beneath the selected machine (`oam_machine_1`) in the Machines pane.
3. Repeat these steps to assign all Managed Servers to their respective machines.

For more information about this screen, see Assign Servers to Machines in *Creating WebLogic Domains Using the Configuration Wizard*.

Virtual Targets

You can skip this screen for Oracle Access Management configuration.

Click **Next** and proceed.

Partitions

The Partitions screen is used to configure partitions for virtual targets in WebLogic Server Multitenant (MT) environments. Select **Next** without selecting any options.

For details about options on this screen, see Partitions in *Creating WebLogic Domains Using the Configuration Wizard*.

> **Note:**
>
> WebLogic Server Multitenant domain partitions are deprecated in WebLogic Server *12.2.1.4.0* and will be removed in the next release.

Configuring Domain Frontend Host

The Domain Frontend Host screen can be used to configure the frontend host for the domain.

Select **Plain** or **SSL** and specify the respective host value.

Click **Next**.

Targeting the Deployments

The Deployments Targeting screen can be used to target the available deployments to the servers.

Make the required modifications, and click **Next**.

Targeting the Services

The Services Targeting screen can be used to target the available services to the Servers.

Make necessary modifications, and click **Next**.

Reviewing Your Configuration Specifications and Configuring the Domain

The Configuration Summary screen shows detailed configuration information for the domain you are about to create.

Review each item on the screen and verify that the information is correct. To make any changes, go back to a screen by clicking the **Back** button or selecting the screen in the navigation pane. Domain creation does not start until you click **Create**.

For more details about options on this screen, see Configuration Summary in *Creating WebLogic Domains Using the Configuration Wizard*.

Writing Down Your Domain Home and Administration Server URL

The End of Configuration screen shows information about the domain you just configured.

Make a note of the following items because you need them later:

- Domain Location
- Administration Server URL

You need the domain location to access scripts that start Node Manager and Administration Server, and you need the URL to access the Administration Server.

Click **Finish** to dismiss the Configuration Wizard.

Updating the System Properties for SSL Enabled Servers

For SSL enabled servers, you must set the required properties in the `setDomainEnv` file in the domain home.

Set the following properties in the *`DOMAIN_HOME`*`/bin/setDomainEnv.sh` (for UNIX) or *`DOMAIN_HOME`*`\bin\setDomainEnv.cmd` (for Windows) file before you start the servers:

- `-Dweblogic.security.SSL.ignoreHostnameVerification=true`
- `-Dweblogic.security.TrustKeyStore=DemoTrust`

Starting the Servers

After a successful configuration, start all processes and servers, including the Administration Server and any Managed Servers.

The components may be dependent on each other so they must be started in the correct order.

> **Note:**
>
> The procedures in this section describe how to start servers and process using the WLST command line or a script. You can also use the Oracle Fusion Middleware Control and the Oracle WebLogic Server Administration Console. See Starting and Stopping Administration and Managed Servers and Node Manager in *Administering Oracle Fusion Middleware.*

To start your Fusion Middleware environment, follow the steps below.

Step 1: Start Node Manager

To start Node Manager, use the `startNodeManager` script:

- (UNIX) *EXISTING_DOMAIN_HOME*`/bin/startNodeManager.sh`
- (Windows) *EXISTING_DOMAIN_HOME*`\bin\startNodeManager.cmd`

Step 2: Start the Administration Server

When you start the Administration Server, you also start the processes running in the Administration Server, including the WebLogic Server Administration Console and Fusion Middleware Control.

To start the Administration Server, use the `startWebLogic` script:

- (UNIX) *EXISTING_DOMAIN_HOME*`/bin/startWebLogic.sh`
- (Windows) *EXISTING_DOMAIN_HOME*`\bin\startWebLogic.cmd`

When you created the domain, if you selected **Production Mode** on the **Domain Mode and JDK** screen, a prompt for the Administrator user login credentials is displayed. Provide the same credentials that you provided on the **Administrator Account** screen.

Note:

For an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), a benign error message may be displayed in the Administration Server logs.
Example message:

```
<AdminServer> <[ACTIVE] ExecuteThread: '63' for queue:
'weblogic.kernel.Default (self-tuning)'> <weblogic> <>
<16023522-e47f-40f4-a66f-7ea3729188d1-00000064>
<1628079696204>
<[severity-value: 8] [rid: 0] [partition-id: 0] [partition-
name: DOMAIN] >
<BEA-240003> <Administration Console encountered the following
error:
java.lang.NoSuchMethodError:
org.glassfish.jersey.internal.LocalizationMessages.WARNING_PROP
ERTIES()Ljava/l ang/String; at
org.glassfish.jersey.internal.config.SystemPropertiesConfigurat
ionModel.getProperties(SystemPropertiesConfigurationModel.java:
122) at
org.glassfish.jersey.internal.config.SystemPropertiesConfigurat
ionProvider.getProperties(SystemPropertiesConfigurationProvider
.java:29) at
org.glassfish.jersey.internal.config.ExternalPropertiesConfigur
ationFactory.readExternalPropertiesMap(ExternalPropertiesConfig
urationFactory.java:55) at
org.glassfish.jersey.internal.config.ExternalPropertiesConfigur
ationFactory.configure(ExternalPropertiesConfigurationFactory.j
ava:72) at
org.glassfish.jersey.internal.config.ExternalPropertiesConfigur
ationFeature.configure(ExternalPropertiesConfigurationFeature.j
ava:26) at
org.glassfish.jersey.model.internal.CommonConfig.configureFeatu
res(CommonConfig.java:730)
```

This error message does not have any functional impact and can be ignored.

Step 3: Start the Managed Servers

- If Node Manager is *not* configured, start the Managed Servers using the following instructions:
  To start a WebLogic Server Managed Server, use the `startManagedWebLogic` script:
  - (UNIX) *EXISTING_DOMAIN_HOME*`/bin/startManagedWebLogic.sh` *managed_server_name admin_url*
  - (Windows) *EXISTING_DOMAIN_HOME*`\bin\startManagedWebLogic.cmd` *managed_server_name admin_url*

When prompted, enter your user name and password. This is the same user name and password which you provided in administrator account screen when creating the domain.

> **Note:**
>
> The startup of a Managed Server will typically start the applications that are deployed to it. Therefore, it should not be necessary to manually start applications after the Managed Server startup.

- If Node Manager is configured, start the Managed Servers using the following instructions:

  1. Launch the Administration Console:
     a. Using a web browser, open the following URL:

        ```
        http://hostname:port/console
        ```

        Where:

        - *hostname* is the administration server host.
        - *port* is the administration server port on which the host server is listening for requests (7001 by default)

     b. When the login page appears, enter the user name and password you used to start the Administration Server.
  2. Start Managed Servers from the Administration Console. For instructions, see Start Managed Servers from the Administration Console.

Verifying the Configuration

After completing all configuration steps, you can perform additional steps to verify that your domain is properly configured.

You can start using the functionality of Oracle Access Management after you successfully configure it. See Getting Started with Oracle Access Management in *Administering Oracle Access Management*.

For information about integrating Oracle Access Management with other Identity Management components, see Introduction to IdM Suite Components Integration in *Integration Guide for Oracle Identity Management Suite*.

For more information about performing additional domain configuration tasks, see Performing Additional Domain Configuration Tasks.

Setting the Memory Parameters for OAM Domain (Optional)

If the initial startup parameter in Oracle Access Management domain, which defines the memory usage, is insufficient, you can increase the value of this parameter.

To change the memory allocation setting, do the following:

1. Edit the *Domain_home*/bin/setUserOverrides.sh file to add the following line:

   ```
   MEM_ARGS="-Xms1024m -Xmx3072m"
   ```

2. Save and close the file.
3. Change the following memory allocation by updating the Java maximum memory allocation pool (Xmx) to `3072m` and initial memory allocation pool (Xms) to `1024m`. For example, change the following line to be:

   ```
   WLS_MEM_ARGS_64BIT="-Xms1024m -Xmx3072m"
   ```

4. Save and close the file.

Updating the java.security File (Optional)

If you wish to integrate Oracle Access Management 12c (12.2.1.4.0) with Oracle Adaptive Access Manager (OAAM) 11*g* Release 2 (11.1.2.3.0), you must update `java.security` file with the following changes, post upgrade:

To do this:

1. Open the `java.security` file located at *JAVA_HOME*/jre/lib/security/ in an editor.
2. Remove `TLSv1`, `TLSv1.1`, `MD5withRSA` from the following key:

   ```
   key - jdk.tls.disabledAlgorithms
   ```

3. Remove `MD5` from the following key:

   ```
   key - jdk.certpath.disabledAlgorithms
   ```

Troubleshooting

This section lists the common issues encountered while configuring Oracle Access Management and their workarounds.

Topics

WADL Generation Does not Show Description

Issue

WADL generation fails and a `java.lang.IllegalStateException: ServiceLocatorImpl` is returned.

```
Exception thrown when provider
class
org.glassfish.jersey.server.internal.monitoring.MonitoringFeature$StatisticsL
istener
was processing MonitoringStatistics. Removing provider from further
processing.
java.lang.IllegalStateException:
ServiceLocatorImpl(__HK2_Generated_6,9,221656053) has been shut down
at
org.jvnet.hk2.internal.ServiceLocatorImpl.checkState(ServiceLocatorImpl.java:
2393)
```

Also, when the WADL generation fails, the description field shows **Root Resource**, instead of a proper description in the following URLs.

```
http://<Host>:<AdminServerPort>/oam/services/rest/11.1.2.0.0/ssa/policyadmin/
application.wadl
http://<Host>:<ManagedServerPort>/iam/access/api/v1/health/application.wadl
```

Resolution

Restart the Admin server and managed servers to resolve the wadl issue.

MDS ReadOnlyStoreException in OAM Policy Manager Diagnostic log

After you configure Oracle Access Management (OAM)12c (12.2.1.4.0), when you start the servers, the following exception is seen in the Administration Server and OAM Policy Manager diagnostic logs:

```
oracle.mds.exception.ReadOnlyStoreException: MDS-01273:
The operation on the  resource /oracle/oam/ui/adfm/DataBindings.cpx
failed because source metadata  store mapped to the namespace / DEFAULT
is read only.
```

This exception does not impact the Administration Console functionality and hence can be safely ignore.

Ignorable Warnings in the Administration Server Logs

After you configure Oracle Access Management 12c (12.2.1.4.0), when you start the Administration Server, the following warning are seen in the Administration Server logs:

```
<Warning>
<oracle.adfinternal.view.faces.renderkit.rich.NavigationPaneRenderer>
<adc2140146> <AdminServer> <[ACTIVE] ExecuteThread: '42' for queue:
'weblogic.kernel.Default (self-tuning)'> <weblogic> <>
<b6ba191d-9c3f-44ce-ad9d-64bd7123baf5-000000e3>
<1502889425767> <[severity-value: 16] [rid: 0] [partition-id: 0]
[partition-name: DOMAIN] >
<BEA-000000> <Warning: There are no items to render for this level>
####<Aug 16, 2017 6:17:06,241 AM PDT> <Warning>
<org.apache.myfaces.trinidad.component.UIXFacesBeanImpl>
```

This has no impact on the functionality, and therefore you can ignore it.

After installing Oracle Access Management, go to Chapter 5: Next Steps After Configuring the Domain.

4

Installing and Configuring the Oracle Identity Governance Software

Follow the steps in this section to install and configure the Oracle Identity Governance software.

> **Note:**
>
> The product Oracle Identity Manager is referred to as Oracle Identity Manager (OIM) and Oracle Identity Governance (OIG) interchangeably in the guide.

- Installing the Oracle Identity Governance Software
- Configuring the Oracle Identity Governance Domain
  After you have installed Oracle Identity Governance, you can configure the domain, which you can also extend for high availability.

Installing the Oracle Identity Governance Software

Follow the steps in this section to install the Oracle Identity Governance software. Before beginning the installation, ensure that you have verified the prerequisites and completed all steps covered in Preparing to Install and Configure.

Oracle Identity Governance 12c (12.2.1.4.0) can be installed by using any of the following methods:

- **Method 1**: Simplified method - by using an quick start installer to install all the products in one go. See Method 1: Simplified Method.
- **Method 2**: Traditional Method - by individually installing the required products. See Method 2: Traditional Method.

For information about supported installation methods, see About Supported Installation Methods.

- Verifying the Installation and Configuration Checklist
  The installation and configuration process requires specific information.
- Verifying the Memory Settings
  To avoid the memory issues for Oracle Identity Manager, ensure that the memory settings are updated as per the requirements.
- Method 1: Simplified Method
- Method 2: Traditional Method
- Verifying the Installation
  After you complete the installation, verify whether it was successful by completing a series of tasks.

Verifying the Installation and Configuration Checklist

The installation and configuration process requires specific information.

Table 4-1 lists important items that you must know before, or decide during, Oracle Identity Governance installation and configuration.

Table 4-1 Installation and Configuration Checklist

Information	Example Value	Description
`JAVA_HOME`	`/home/Oracle/Java/jdk1.8.0_211`	Environment variable that points to the Java JDK home directory.
Database host	`examplehost.exampledomain`	Name and domain of the host where the database is running.
Database port	`1521`	Port number that the database listens on. The default Oracle database listen port is `1521`.
Database service name	`orcl.exampledomain`	Oracle databases require a unique service name. The default service name is `orcl`.
DBA username	`SYS`	Name of user with database administration privileges. The default DBA user on Oracle databases is `SYS`.
DBA password	`myDBApw957`	Password of the user with database administration privileges.
`ORACLE_HOME`	`/home/Oracle/`*`product`*`/ORACLE_HOME`	Directory in which you will install your software. This directory will include Oracle Fusion Middleware Infrastructure, Oracle SOA Suite, and Oracle Identity Governance, as needed.
WebLogic Server hostname	`examplehost.exampledomain`	Host name for Oracle WebLogic Server and Oracle Identity Governance consoles.
Console port	`7001`	Port for Oracle WebLogic Server and Oracle Identity Governance consoles.
`DOMAIN_HOME`	`/home/Oracle/config/domains/idm_domain`	Location in which your domain data is stored.
`APPLICATION_HOME`	`/home/Oracle/config/applications/idm_domain`	Location in which your application data is stored.
Administrator user name for your WebLogic domain	`weblogic`	Name of the user with Oracle WebLogic Server administration privileges. The default administrator user is `weblogic`.

Table 4-1 (Cont.) Installation and Configuration Checklist

Information	Example Value	Description
Administrator user password	`myADMpw902`	Password of the user with Oracle WebLogic Server administration privileges.
RCU	*`ORACLE_HOME`*`/oracle_common/bin`	Path to the Repository Creation Utility (RCU).
RCU schema prefix	`oim`	Prefix for names of database schemas used by Oracle Identity Governance.
RCU schema password	`myRCUpw674`	Password for the database schemas used by Oracle Identity Governance.
Configuration utility	*`ORACLE_HOME`*`/oracle_common/common/bin`	Path to the Configuration Wizard for domain creation and configuration.

Verifying the Memory Settings

To avoid the memory issues for Oracle Identity Manager, ensure that the memory settings are updated as per the requirements.

On Linux, do the following:

1. Ensure that you set the following parameters in the `/etc/security/limits.conf` file, to the specified values:
   - `FUSION_USER_ACCOUNT soft nofile 32767`
   - `FUSION_USER_ACCOUNT hard nofile 327679`
2. Ensure that you set `UsePAM` to `Yes` in the /etc/ssh/sshd_config file.
3. Restart `sshd`.
4. Log out (or reboot) and log in to the system again.

> **Note:**
>
> Before you start the Oracle Identity Governance 12c (12.2.1.4.0) Server, post configuration, run the following command to increase the limit of open files, so that you do not run into memory issues:
> `limit maxproc 16384`

Method 1: Simplified Method

You can install the Oracle Identity Governance software by using a quickstart installer. For Oracle Identity Governance a quickstart installer is available, which installs Infrastructure, Oracle SOA Suite, and Oracle Identity Governance 12c (12.2.1.4.0) in one go. You do not have to install these softwares using separate installers.

- Roadmap for Installing and Configuring Oracle Identity Governance Using Simplified Installation
  Use the roadmap provided in this section to install and configure Oracle Identity Governance (OIG) using the simplified installation process.
- Installing Oracle Identity Governance Using Quickstart Installer
  Complete the instructions in this section to install Oracle Identity Governance.

Roadmap for Installing and Configuring Oracle Identity Governance Using Simplified Installation

Use the roadmap provided in this section to install and configure Oracle Identity Governance (OIG) using the simplified installation process.

This table provides the high-level steps for installing and configuring Oracle Identity Governance.

Table 4-2 Task Roadmap for Installing and Configuring Oracle Identity Governance Using Simplified Installation

Task	Description
Verify if your system meets the minimum hardware and software requirements.	See Roadmap for Verifying Your System Environment
Install Oracle Fusion Middleware Infrastructure, Oracle SOA Suite, and Oracle Identity Governance 12.2.1.4.0 using the quickstart installer. This task involves obtaining the quickstart installer, starting the installation program, and navigating the installer screens.	See Installing Oracle Identity Governance Using Quickstart Installer
Create the database schemas using Repository Creation Utility (RCU).	See Creating the Database Schemas
Configure and update the Oracle Identity Governance domain.	See Configuring and updating the OIG domain
Perform the necessary post-configuration tasks.	See Performing Post-Configuration Tasks
Start the Node Manager, Administration Server, Oracle SOA Suite Managed Server, and the OIG Managed Server.	See Starting the Servers
Integrate Oracle Identity Governance with Oracle SOA Suite.	See Integrating Oracle Identity Governance with Oracle SOA Suite
Verify the configuration.	See Verifying the Configuration
Refer to the bootstrap report for the configuration details and for any issues or warnings thrown during the installation process.	See Analyzing the Bootstrap Report
Access the Oracle Identity Governance Design Console, if required.	See Installing and Accessing the Oracle Identity Governance Design Console

Installing Oracle Identity Governance Using Quickstart Installer

Complete the instructions in this section to install Oracle Identity Governance.

Topics:

- Obtaining the Quickstart Installer
  Obtain the quickstart installer distribution on *Technical Resources from Oracle*.
- Starting the Quickstart Installation Program
  Start the quickstart installation program by running the java executable from the JDK directory.
- Navigating the Quickstart Installation Screens
  The quickstart installer shows a series of screens where you verify or enter information.

Obtaining the Quickstart Installer

Obtain the quickstart installer distribution on *Technical Resources from Oracle*.

See Obtaining Product Distributions in *Planning an Installation of Oracle Fusion Middleware*.

After downloading the required `.zip` file, unzip the `.zip` file to obtain the `.jar` distributions.

> **Note:**
>
> No prerequisite software is required for `qstart`.

Starting the Quickstart Installation Program

Start the quickstart installation program by running the java executable from the JDK directory.

> **Note:**
>
> Before running the quickstart installation program, you must verify the supported JDK version is installed.

Run the following command from the JDK directory:

- On UNIX:

  ```
  $JAVA_HOME/bin/java –jar fmw_12.2.1.4.0_idmquickstart.jar
  ```

- On Windows:

  ```
  $JAVA_HOME\bin\java –jar fmw_12.2.1.4.0_idmquickstart.jar
  ```

Navigating the Quickstart Installation Screens

The quickstart installer shows a series of screens where you verify or enter information.

The following table lists the order in which installer screens appear. If you need additional help with an installation screen, click **Help**.

Table 4-3 Oracle Identity Governance Quickstart Install Screens

Screen	Description
Installation Inventory Setup	On Linux or UNIX operating systems, this screen opens if this is the first time you are installing any Oracle product on this host. Specify the location where you want to create your central inventory. Make sure that the operating system group name selected on this screen has write permissions to the central inventory location. See About the Oracle Central Inventory in *Installing Software with the Oracle Universal Installer*. This screen does not appear on Windows operating systems.
Welcome	Review the information to make sure that you have met all the prerequisites, then click **Next**.
Auto Updates	Select to skip automatic updates, select patches, or search for the latest software updates, including important security updates, through your My Oracle Support account.
Installation Location	Specify your Oracle home directory location. You can click **View** to verify and ensure that you are installing the products in the correct Oracle home.
Prerequisite Checks	This screen verifies that your system meets the minimum necessary requirements. To view the list of tasks that gets verified, select **View Successful Tasks**. To view log details, select **View Log**. If any prerequisite check fails, then an error message appears at the bottom of the screen. Fix the error and click **Rerun** to try again. To ignore the error or the warning message and continue with the installation, click **Skip** (not recommended).
Installation Summary	Use this screen to verify installation options you selected. If you want to save these options to a response file, click **Save Response File** and enter the response file location and name. The response file collects and stores all the information that you have entered, and enables you to perform a silent installation (from the command line) at a later time. Click **Install** to begin the installation.
Installation Progress	This screen shows the installation progress. When the progress bar reaches 100% complete, click **Finish** to dismiss the installer, or click **Next** to see a summary.
Installation Complete	This screen displays the Installation Location and the Feature Sets that are installed. Review this information and click **Finish** to close the installer.

After completing installation by using the simplified method, proceed to complete the following:

1. Verifying the Installation
2. Configuring the Oracle Identity Governance Domain

Method 2: Traditional Method

You can install the Oracle Identity Governance software in traditional method, by individually installing required products.
Dependant Softwares for installing Oracle Identity Governance in traditional method:

> **Note:**
>
> Install products in the specified order.

1. Oracle Fusion Middleware Infrastructure 12c (12.2.1.4.0)
2. Oracle SOA Suite 12c (12.2.1.4.0)

For information about installing Oracle Fusion Middleware Infrastructure 12c (12.2.1.4.0), see Installing the Infrastructure Softwarein *Installing and Configuring the Oracle Fusion Middleware Infrastructure*.

For information about installing Oracle SOA Suite 12c (12.2.1.4.0), see Installing the Oracle SOA Suite and Oracle Business Process Management Software in *Installing and Configuring Oracle SOA Suite and Business Process Management*.

- Starting the Installation Program
  Before running the installation program, you must verify the JDK and prerequisite software is installed.
- Navigating the Installation Screens
  The installer shows a series of screens where you verify or enter information.

Starting the Installation Program

Before running the installation program, you must verify the JDK and prerequisite software is installed.

To start the installation program:

1. Sign in to the host system.
2. Change to the directory where you downloaded the installation program.
3. You must have installed the Oracle Fusion Middleware Infrastructure 12c (12.2.1.4.0). For instructions, see Installing the Infrastructure Software in *Installing and Configuring the Oracle Fusion Middleware Infrastructure*.
4. You must have installed the Oracle SOA Suite 12c (12.2.1.4.0). For instructions, see Installing the Oracle SOA Suite and Oracle Business Process Management Software in *Installing and Configuring Oracle SOA Suite and Business Process Management*.

> **Note:**
>
> When installing Oracle SOA Suite 12c (12.2.1.4.0), in the Installation Type screen, select the **SOA Suite** option.

5. Start the installation program by running the `java` executable from the JDK directory. For example:
   - (UNIX) `/home/Oracle/Java/jdk1.8.0_211/bin/java -jar fmw_12.2.1.4.0_idm.jar`
   - (Windows) `C:\home\Oracle\Java\jdk1.8.0_211\bin\java -jar fmw_12.2.1.4.0_idm.jar`

> **Note:**
>
> You can also start the installer in silent mode using a saved response file instead of launching the installer screens. For more about silent or command line installation, see Using the Oracle Universal Installer in Silent Mode in *Installing Software with the Oracle Universal Installer*.

When the installation program appears, you are ready to begin the installation.

Navigating the Installation Screens

The installer shows a series of screens where you verify or enter information.

The following table lists the order in which installer screens appear. If you need additional help with an installation screen, click **Help**.

Table 4-4 Install Screens

Screen	Description
Installation Inventory Setup	On Linux or UNIX operating systems, this screen opens if this is the first time you are installing any Oracle product on this host. Specify the location where you want to create your central inventory. Make sure that the operating system group name selected on this screen has write permissions to the central inventory location. See About the Oracle Central Inventory in *Installing Software with the Oracle Universal Installer*. This screen does not appear on Windows operating systems.
Welcome	Review the information to make sure that you have met all the prerequisites, then click **Next**.
Auto Updates	Select to skip automatic updates, select patches, or search for the latest software updates, including important security updates, through your My Oracle Support account.

Table 4-4 (Cont.) Install Screens

Screen	Description
Installation Location	Specify your Oracle home directory location. This Oracle home must include Oracle Fusion Middleware Infrastructure 12c (12.2.1.4.0). You can click **View** to verify and ensure that you are installing in the correct Oracle home. **Note:** Ensure that the Oracle Home path does not contain space.
Installation Type	Use the **Collocated** Installation Type. Collocated mode is a type of installation that is managed through WebLogic Server. To install in collocated mode, you must have installed the required dependant softwares.
Prerequisite Checks	This screen verifies that your system meets the minimum necessary requirements. To view the list of tasks that gets verified, select **View Successful Tasks**. To view log details, select **View Log**. If any prerequisite check fails, then an error message appears at the bottom of the screen. Fix the error and click **Rerun** to try again. To ignore the error or the warning message and continue with the installation, click **Skip** (not recommended).
Installation Summary	Use this screen to verify installation options you selected. If you want to save these options to a response file, click **Save Response File** and enter the response file location and name. The response file collects and stores all the information that you have entered, and enables you to perform a silent installation (from the command line) at a later time. Click **Install** to begin the installation.
Installation Progress	This screen shows the installation progress. When the progress bar reaches 100% complete, click **Finish** to dismiss the installer, or click **Next** to see a summary.
Installation Complete	This screen displays the Installation Location and the Feature Sets that are installed. Review this information and click **Finish** to close the installer.

Verifying the Installation

After you complete the installation, verify whether it was successful by completing a series of tasks.

- Reviewing the Installation Log Files
  Review the contents of the installation log files to make sure that the installer did not encounter any problems.
- Checking the Directory Structure
  The contents of your installation vary based on the options that you selected during the installation.

- Viewing the Contents of the Oracle Home
  You can view the contents of the Oracle home directory by using the `viewInventory` script.

Reviewing the Installation Log Files

Review the contents of the installation log files to make sure that the installer did not encounter any problems.

By default, the installer writes logs files to the *`Oracle_Inventory_Location`*`/logs` (on UNIX operating systems) or *`Oracle_Inventory_Location`*`\logs` (on Windows operating systems) directory.

For a description of the log files and where to find them, see Installation Log Files in *Installing Software with the Oracle Universal Installer*.

Checking the Directory Structure

The contents of your installation vary based on the options that you selected during the installation.

See What Are the Key Oracle Fusion Middleware Directories? in *Understanding Oracle Fusion Middleware*.

Viewing the Contents of the Oracle Home

You can view the contents of the Oracle home directory by using the `viewInventory` script.

See Viewing the Contents of an Oracle Home in *Installing Software with the Oracle Universal Installer*.

Configuring the Oracle Identity Governance Domain

After you have installed Oracle Identity Governance, you can configure the domain, which you can also extend for high availability.

> **Note:**
>
> In this document, the variable *`OIM_HOME`* is used for *`ORACLE_HOME`*`/idm` (Unix) and *`ORACLE_HOME`*`\idm` (Windows).

Refer to the following sections to create the database schemas, configure a WebLogic domain, and verify the configuration:

- Creating the Database Schemas
  Before you can configure an Oracle Identity Governance domain, you must install required schemas on a certified database for use with this release of Oracle Fusion Middleware.
- Configuring the Domain
  Use the Configuration Wizard to create and configure a domain.

- Performing Post-Configuration Tasks
  After you configure the Oracle Identity Governance domain, perform the necessary post-configuration tasks.
- Starting the Servers
  After a successful configuration, start all processes and servers, including the Administration Server and any Managed Servers.
- Integrating Oracle Identity Governance with Oracle SOA Suite
  If you wish to integrate Oracle Identity Governance with Oracle SOA Suite, use the Enterprise Manager console to do the same.
- Verifying the Configuration
  After completing all configuration steps, you can perform additional steps to verify that your domain is properly configured.
- Analyzing the Bootstrap Report
  When you start the Oracle Identity Governance server, the bootstrap report is generated at *DOMAIN_HOME*/servers/*oim_server1*/logs/BootStrapReportPreStart.html.
- Installing and Accessing the Oracle Identity Governance Design Console
  If you wish to set up only the Oracle Identity Governance Design Console in a machine where OIG server is not configured, then you must install Oracle Identity Governance 12c (12.2.1.4.0) in standalone mode, and then invoke the Design Console.
- Troubleshooting
  This section lists the common issues encountered while configuring Oracle Identity Governance and their workarounds.

Creating the Database Schemas

Before you can configure an Oracle Identity Governance domain, you must install required schemas on a certified database for use with this release of Oracle Fusion Middleware.

- Installing and Configuring a Certified Database
  Before you create the database schemas, you must install and configure a certified database, and verify that the database is up and running.
- Starting the Repository Creation Utility
  Start the Repository Creation Utility (RCU) after you verify that a certified JDK is installed on your system.
- Navigating the Repository Creation Utility Screens to Create Schemas
  Enter required information in the RCU screens to create the database schemas.

Installing and Configuring a Certified Database

Before you create the database schemas, you must install and configure a certified database, and verify that the database is up and running.

> **Note:**
>
> For an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), you must modify the wallet settings, set the environment variables, and apply patches on `ORACLE HOME`. For more information, see Settings to connect to an Autonomous Transaction Processing Database and Applying Patches on ORACLE HOME.

See About Database Requirements for an Oracle Fusion Middleware Installation.

Starting the Repository Creation Utility

Start the Repository Creation Utility (RCU) after you verify that a certified JDK is installed on your system.

To start the RCU:

1. Verify that a certified JDK already exists on your system by running `java -version` from the command line. For 12c (12.2.1.4.0), the certified JDK is 1.8.0_211 and later.

   See About JDK Requirements for an Oracle Fusion Middleware Installation.
2. Ensure that the `JAVA_HOME` environment variable is set to the location of the certified JDK. For example:
   - (UNIX) `setenv JAVA_HOME /home/Oracle/Java/jdk1.8.0_211`
   - (Windows) `set JAVA_HOME=C:\home\Oracle\Java\jdk1.8.0_211`
3. Change to the following directory:
   - (UNIX) *`ORACLE_HOME`*`/oracle_common/bin`
   - (Windows) *`ORACLE_HOME`*`\oracle_common\bin`
4. Enter the following command:
   - (UNIX) `./rcu`
   - (Windows) `rcu.bat`

Navigating the Repository Creation Utility Screens to Create Schemas

Enter required information in the RCU screens to create the database schemas.

- Introducing the RCU
  The Welcome screen is the first screen that appears when you start the RCU.

- Selecting a Method of Schema Creation
  Use the Create Repository screen to select a method to create and load component schemas into the database.
- Providing Database Connection Details
  On the Database Connection Details screen, provide the database connection details for the RCU to connect to your database.
- Specifying a Custom Prefix and Selecting Schemas
- Specifying Schema Passwords
  On the Schema Passwords screen, specify how you want to set the schema passwords on your database, then enter and confirm your passwords.
- Specifying Custom Variables
- Completing Schema Creation
  Navigate through the remaining RCU screens to complete schema creation.

Introducing the RCU

The Welcome screen is the first screen that appears when you start the RCU.

Click **Next**.

Selecting a Method of Schema Creation

Use the Create Repository screen to select a method to create and load component schemas into the database.

On the Create Repository screen:

- If you have the necessary permissions and privileges to perform DBA activities on your database, select **System Load and Product Load**. This procedure assumes that you have SYSDBA privileges.
- If you do *not* have the necessary permissions or privileges to perform DBA activities in the database, you must select **Prepare Scripts for System Load** on this screen. This option generates a SQL script that you can give to your database administrator. See About System Load and Product Load in *Creating Schemas with the Repository Creation Utility*.
- If the DBA has already run the SQL script for System Load, select **Perform Product Load**.

> **Note:**
>
> For an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), you must create schemas as a `Normal` user, and though, you do not have full SYS or SYSDBA privileges on the database, you must select **System Load and Product Load**.

Providing Database Connection Details

On the Database Connection Details screen, provide the database connection details for the RCU to connect to your database.

> **Note:**
>
> If you are unsure of the service name for your database, you can obtain it from the `SERVICE_NAMES` parameter in the initialization parameter file of the database. If the initialization parameter file does not contain the `SERVICE_NAMES` parameter, then the service name is the same as the global database name, which is specified in the `DB_NAME` and `DB_DOMAIN` parameters.
>
> For an Oracle Autonomous Transaction Processing-Shared (ATP-S) database, you must use only one of the database service names, `<databasename>_tpurgent` or `<databasename>_tp`, specified in `tnsnames.ora`. For database service name details, see Database Service Names for Autonomous Transaction Processing and Autonomous JSON Database.
>
> To create schemas on an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), you can specify the connection credentials using only the **Connection String** option. In this screen, a warning message is displayed. You can ignore the warning and continue with the schema creation. For more information, see SYS DBA Privileges Warning After Applying Patches.

For example:

Database Type: `Oracle Database`
Connection String Format: `Either one of the format one can select`

- If you choose connection parameter, fill the following details:

  Host Name: `examplehost.exampledomain.com`
  Port: `1521`
  Service Name: `Orcl.exampledomain.com`
  User Name: `sys`
  Password: `******`
  Role: `SYSDBA`

- If you choose connection string, fill the following details:

  Connection String:
  `examplehost.exampledomain.com:1521:Orcl.exampledomain.com`
  User Name: `sys`
  Password: `******`
  Role: `SYSDBA`

For an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), enter connect string in the following format:

```
jdbc:oracle:thin:@TNS_alias?TNS_ADMIN=<path of the wallet files,
ojdbc.properties, and tnsnames.ora>
```

In the connect string, you must pass `TNS_alias` as the database name found in `tnsnames.ora`, and `TNS_ADMIN` property to the location of the wallet files, `ojdbc.properties`, and `tnsnames.ora`.

> **Note:**
>
> For an Oracle Autonomous Transaction Processing-Shared (ATP-S) database, you must use only one of the database service names, `<databasename>_tpurgent` or `<databasename>_tp`, specified in `tnsnames.ora`. For database service name details, see Database Service Names for Autonomous Transaction Processing and Autonomous JSON Database.

Example connect string for Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database:

```
jdbc:oracle:thin:@dbname_medium?TNS_ADMIN=/users/test/wallet_dbname/
```

Example connect string for Oracle Autonomous Transaction Processing-Shared (ATP-S) database:

```
jdbc:oracle:thin:@dbname_tp?TNS_ADMIN=/users/test/wallet_dbname/
```

Click **Next** to proceed, then click **OK** in the dialog window that confirms a successful database connection.

Specifying a Custom Prefix and Selecting Schemas

Select **Create new prefix**, specify a custom prefix, then expand **IDM Schemas** and select the **Oracle Identity Manager** schema. This action automatically selects the following schemas as dependencies:

- User Messaging Service (UMS)
- Metadata Services (MDS)
- Oracle Platform Security Services (OPSS)
- Audit Services (IAU)
- Audit Services Append (IAU_Append)
- Audit Services Viewer (IAU_Viewer)
- WebLogic Services (WLS)
- Common Infrastructure Services (STB)
- SOA Infrastructure (SOAINFRA)

The schema Common Infrastructure Services (STB) is automatically created. This schema is dimmed; you cannot select or deselect it. This schema enables you to retrieve information from RCU during domain configuration. For more information, see "Understanding the Service Table Schema" in *Creating Schemas with the Repository Creation Utility*.

The custom prefix is used to logically group these schemas together for use in this domain only; you must create a unique set of schemas for each domain. Schema sharing across domains is not supported.

> **Tip:**
>
> For more information about custom prefixes, see "Understanding Custom Prefixes" in *Creating Schemas with the Repository Creation Utility*.
>
> For more information about how to organize your schemas in a multi-domain environment, see "Planning Your Schema Creation" in *Creating Schemas with the Repository Creation Utility*.

> **Tip:**
>
> You must make a note of the custom prefix you choose to enter here; you will need this later on during the domain creation process.

Click **Next** to proceed, then click **OK** on the dialog window confirming that prerequisite checking for schema creation was successful.

Specifying Schema Passwords

On the Schema Passwords screen, specify how you want to set the schema passwords on your database, then enter and confirm your passwords.

> **Note:**
>
> For an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), the schema password must be minimum 12 characters, and must contain at least one uppercase, one lower case, and one number.

You must make a note of the passwords you set on this screen; you will need them later on during the domain creation process.

Click **Next**.

Specifying Custom Variables

On the Custom Variables screen, accept the default values and click **Next**.

> **Tip:**
>
> For more information about options on this screen, see Custom Variables in *Creating Schemas with the Repository Creation Utility*.

Completing Schema Creation

Navigate through the remaining RCU screens to complete schema creation.

On the Map Tablespaces screen, the Encrypt Tablespace check box appears *only* if you enabled Transparent Data Encryption (TDE) in the database (Oracle or Oracle EBR) when you start the RCU.

To complete schema creation:

1. On the Map Tablespaces screen, select **Encrypt Tablespace** if you want to encrypt all new tablespaces that the RCU creates.
2. In the Completion Summary screen, click **Close** to dismiss the RCU.

   For an Oracle Autonomous Transaction Processing-Shared (ATP-S) database, in the **Map Tablespaces** screen you must override the default tablespaces and the temporary tablespaces, and also override the additional tablespaces, if applicable. See Map Tablespaces.

   If you encounter any issues when you create schemas on an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), see Troubleshooting Tips for Schema Creation on an Autonomous Transaction Processing Database in *Creating Schemas with the Repository Creation Utility* and Issues Related to Product Installation and Configuration on an Autonomous Database in *Release Notes for Oracle Fusion Middleware Infrastructure*.

Configuring the Domain

Use the Configuration Wizard to create and configure a domain.

For information on other methods to create domains, see Additional Tools for Creating, Extending, and Managing WebLogic Domains in *Creating WebLogic Domains Using the Configuration Wizard*.

- Starting the Configuration Wizard
  Start the Configuration Wizard to begin configuring a domain.
- Navigating the Configuration Wizard Screens to Create and Configure the Domain
  Enter required information in the Configuration Wizard screens to create and configure the domain for the topology.

Starting the Configuration Wizard

Start the Configuration Wizard to begin configuring a domain.

To start the Configuration Wizard:

1. Change to the following directory:

   (UNIX) *ORACLE_HOME*/oracle_common/common/bin

(Windows) *ORACLE_HOME*\oracle_common\common\bin

where *ORACLE_HOME* is your 12c (12.2.1.4.0) Oracle home.

2. Enter the following command:

   (UNIX) `./config.sh`

   (Windows) `config.cmd`

Navigating the Configuration Wizard Screens to Create and Configure the Domain

Enter required information in the Configuration Wizard screens to create and configure the domain for the topology.

> **Note:**
>
> You can use this procedure to extend an existing domain. If your needs do not match the instructions in the procedure, be sure to make your selections accordingly, or see the supporting documentation for more details.

- Selecting the Domain Type and Domain Home Location
  Use the Configuration Type screen to select a Domain home directory location, optimally outside the Oracle home directory.
- Selecting the Configuration Templates for Oracle Identity Manager
- Configuring High Availability Options
  If you are not using a high availability setup, accept the default values on this screen and then click **Next** to proceed to the next screen. Use this screen to configure service migration and persistence settings that affect high availability.
- Selecting the Application Home Location
  Use the Application Location screen to select the location to store applications associated with your domain, also known as the *Application home* directory.
- Configuring the Administrator Account
  Use the Administrator Account screen to specify the user name and password for the default WebLogic Administrator account for the domain.
- Specifying the Domain Mode and JDK
  Use the Domain Mode and JDK screen to specify the domain mode and Java Development Kit (JDK).
- Specifying the Database Configuration Type
  Use the Database Configuration type screen to specify details about the database and database schema.
- Specifying JDBC Component Schema Information
  Use the JDBC Component Schema screen to verify or specify details about the database schemas.
- Testing the JDBC Connections
  Use the JDBC Component Schema Test screen to test the data source connections.

- Entering Credentials
  Use the Credentials screen to set credentials for each key in the domain.
- Specifying the Path to the Keystore Certificate or Key
- Selecting Advanced Configuration
  Use the Advanced Configuration screen to complete the domain configuration.
- Configuring the Administration Server Listen Address
  Use the Administration Server screen to select the IP address of the host.
- Configuring Node Manager
  Use the Node Manager screen to select the type of Node Manager you want to configure, along with the Node Manager credentials.
- Configuring Managed Servers for Oracle Identity Manager
- Configuring a Cluster for Oracle Identity Manager
  Use the Clusters screen to create a new cluster. This is required for an Oracle Identity Governance high availability setup.
- Defining Server Templates
  If you are creating dynamic clusters for a high availability setup, use the Server Templates screen to define one or more server templates for domain.
- Configuring Dynamic Servers
  If you are creating dynamic clusters for a high availability setup, use the Dynamic Servers screen to configure the dynamic servers.
- Assigning Oracle Identity Manager Managed Servers to the Cluster
  If you are configuring a single-node non-clustered setup, click **Next** and go to next screen. Use the Assign Servers to Clusters screen to assign Managed Servers to a new *configured cluster*. A configured cluster is a cluster you configure manually. You do not use this screen if you are configuring a *dynamic cluster*, a cluster that contains one or more generated server instances that are based on a server template.
- Configuring Coherence Clusters
  Use the Coherence Clusters screen to configure the Coherence cluster.
- Creating a New Oracle Identity Manager Machine
  Use the Machines screen to create new machines in the domain. A machine is required so that Node Manager can start and stop servers.
- Assigning Servers to Oracle Identity Manager Machines
  Use the Assign Servers to Machines screen to assign the Administration Server and Managed Servers to the new machine you just created.
- Virtual Targets
  If you have a WebLogic Server Multitenant (MT) environment, you use the Virtual Targets screen to add or delete virtual targets. For this installation (not a WebLogic Server MT environment), you do not enter any values; just select **Next**.
- Partitions
  The Partitions screen is used to configure partitions for virtual targets in WebLogic Server Multitenant (MT) environments. Select **Next** without selecting any options.
- Configuring Domain Frontend Host
  The Domain Frontend Host screen can be used to configure the frontend host for the domain.
- Targeting the Deployments
  The Deployments Targeting screen can be used to target the available deployments to the servers.

- Targeting the Services
  The Services Targeting screen can be used to target the available services to the Servers.
- File Stores
  The File Stores screen lists the available file stores.
- Reviewing Your Configuration Specifications and Configuring the Domain
  The Configuration Summary screen shows detailed configuration information for the domain you are about to create.
- Writing Down Your Domain Home and Administration Server URL
  The End of Configuration screen shows information about the domain you just configured.
- Additional Domain Configuration
  Use the Configuration Wizard to update the domain.

Selecting the Domain Type and Domain Home Location

Use the Configuration Type screen to select a Domain home directory location, optimally outside the Oracle home directory.

Oracle recommends that you locate your Domain home in accordance with the directory structure in What Are the Key Oracle Fusion Middleware Directories? in *Understanding Oracle Fusion Middleware*, where the Domain home is located outside the Oracle home directory. This directory structure helps avoid issues when you need to upgrade or reinstall software.

> **Note:**
>
> Use different *domain_homes* for Oracle Access Management and Oracle Identity Governance.

To specify the Domain type and Domain home directory:

1. On the Configuration Type screen, select **Create a new domain**.
2. In the Domain Location field, specify your Domain home directory.

For more details about this screen, see Configuration Type in *Creating WebLogic Domains Using the Configuration Wizard*.

Selecting the Configuration Templates for Oracle Identity Manager

On the Templates screen, make sure **Create Domain Using Product Templates** is selected, then select the **Oracle Identity Manager** template.

Selecting this template automatically selects the following as dependencies:

- Oracle Enterprise Manager
- Oracle WSM Policy Manager
- Oracle JRF
- WebLogic Coherence Cluster Extension

> **Note:**
>
> - The basic WebLogic domain is pre-selected.
> - Do not select Oracle SOA Suite in this screen. Oracle SOA Suite is automatically configured.

More information about the options on this screen can be found in Templates in *Creating WebLogic Domains Using the Configuration Wizard*.

Configuring High Availability Options

If you are not using a high availability setup, accept the default values on this screen and then click **Next** to proceed to the next screen. Use this screen to configure service migration and persistence settings that affect high availability.

This screen appears for the first time when you create a cluster that uses automatic service migration, persistent stores, or both, and all subsequent clusters that are added to the domain by using the Configuration Wizard, automatically apply the selected HA options.

Enable Automatic Service Migration

Select **Enable Automatic Service Migration** to enable pinned services to migrate automatically to a healthy Managed Server for failover. It configures migratable target definitions that are required for automatic service migration and the cluster leasing. Choose one of these cluster leasing options:

- Database Leasing - Managed Servers use a table on a valid JDBC System Resource for leasing. Requires that the Automatic Migration data source have a valid JDBC System Resource. If you select this option, the Migration Basis is configured to Database and the Data Source for Automatic Migration is also automatically configured by the Configuration Wizard. If you have a high availability database, such as Oracle RAC, to manage leasing information, configure the database for server migration according to steps in High-availability Database Leasing.
- Consensus Leasing - Managed Servers maintain leasing information in-memory. You use Node Manager to control Managed Servers in a cluster. (All servers that are migratable, or which could host a migratable target, must have a Node Manager associated with them.) If you select this option, the Migration Basis is configured to Consensus by the Configuration Wizard.

See Leasing for more information on leasing.

See Service Migration for more information on Automatic Service Migration.

JTA Transaction Log Persistence

This section has two options: **Default Persistent Store** and **JDBC TLog Store**.

- Default Persistent Store - Configures the JTA Transaction Log store of the servers in the default file store.
- JDBC TLog Store - Configures the JTA Transaction Log store of the servers in JDBC stores.

Oracle recommends that you select **JDBC TLog Store**. When you complete the configuration, you have a cluster where JDBC persistent stores are set up for Transaction logs.

For more details on persistent and TLOG stores, see the following topics in *Developing JTA Applications for Oracle WebLogic Server*:

- Using the Default Persistent Store
- Using a JDBC TLOG Store

JMS Server Persistence

A persistent **JMS store** is a physical repository for storing persistent message data and durable subscribers. It can be either a disk-based **file store** or a JDBC-accessible database. You can use a **JMS file store** for paging of messages to disk when memory is exhausted.

- JMS File Store - Configures a component to use JMS File Stores. If you select this option, you can choose the **File Store** option in the Advanced Configuration Screen to change the settings, if required. In the File Stores screen, you can set file store names, directories, and synchronous write policies.
- JMS JDBC Store - Configures a component to use JDBC stores for all its JMS servers. When you complete the configuration, you have a cluster and JDBC persistent stores are configured for the JMS servers.

  This is the recommended option for Oracle Identity Governance 12c (12.2.1.4.0).

Selecting the Application Home Location

Use the Application Location screen to select the location to store applications associated with your domain, also known as the *Application home* directory.

Oracle recommends that you locate your Application home in accordance with the directory structure in What Are the Key Oracle Fusion Middleware Directories? in *Understanding Oracle Fusion Middleware*, where the Application home is located outside the Oracle home directory. This directory structure helps avoid issues when you need to upgrade or re-install your software.

For more about the Application home directory, see About the Application Home Directory.

For more information about this screen, see Application Location in *Creating WebLogic Domains Using the Configuration Wizard*.

Configuring the Administrator Account

Use the Administrator Account screen to specify the user name and password for the default WebLogic Administrator account for the domain.

Oracle recommends that you make a note of the user name and password that you enter on this screen; you need these credentials later to boot and connect to the domain's Administration Server.

For more information about this screen, see Administrator Account in *Creating WebLogic Domains Using the Configuration Wizard.*

Specifying the Domain Mode and JDK

Use the Domain Mode and JDK screen to specify the domain mode and Java Development Kit (JDK).

On the Domain Mode and JDK screen:

- Select **Production** in the **Domain Mode** field.
- Select the **Oracle HotSpot JDK** in the **JDK** field.

For more information about this screen, see Domain Mode and JDK in *Creating WebLogic Domains Using the Configuration Wizard*.

Specifying the Database Configuration Type

Use the Database Configuration type screen to specify details about the database and database schema.

On the Database Configuration type screen, select **RCU Data**. This option instructs the Configuration Wizard to connect to the database and Service Table (STB) schema to automatically retrieve schema information for schemas needed to configure the domain.

> **Note:**
>
> If you select **Manual Configuration** on this screen, you must manually fill in parameters for your schema on the next screen.
>
> For an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), you must select only the **RCU Data** option.

After selecting **RCU Data**, specify details in the following fields:

Field	Description
DBMS/Service	Enter the database DBMS name, or service name if you selected a service type driver. Example: `orcl.exampledomain.com`
Host Name	Enter the name of the server hosting the database. Example: `examplehost.exampledomain.com`
Port	Enter the port number on which the database listens. Example: `1521`
Schema Owner Schema Password	Enter the username and password for connecting to the database's Service Table schema. This is the schema username and password entered for the Service Table component on the Schema Passwords screen in the RCU (see Specifying Schema Passwords). The default username is *`prefix`*`_STB`, where *`prefix`* is the custom prefix that you defined in the RCU.

For an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared

(ATP-S)), specify the connection credentials using only the **Connection URL String** option, and enter the connect string in the following format:

```
jdbc:oracle:thin:@TNS_alias?TNS_ADMIN=<path of the wallet files,
ojdbc.properties, and tnsnames.ora>
```

In the connect string, you must pass `TNS_alias` as the database name found in `tnsnames.ora`, and `TNS_ADMIN` property to the location of the wallet files, `ojdbc.properties`, and `tnsnames.ora`.

Example connect string for Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database:

```
jdbc:oracle:thin:@dbname_medium?TNS_ADMIN=/users/test/wallet_dbname/
```

Example connect string for Oracle Autonomous Transaction Processing-Shared (ATP-S) database:

```
jdbc:oracle:thin:@dbname_tp?TNS_ADMIN=/users/test/wallet_dbname/
```

Click **Get RCU Configuration** when you finish specifying the database connection information. The following output in the Connection Result Log indicates that the operation succeeded:

```
Connecting to the database server...OK
Retrieving schema data from database server...OK
Binding local schema components with retrieved data...OK

Successfully Done.
```

For more information about the schema installed when the RCU is run, see About the Service Table Schema in *Creating Schemas with the Repository Creation Utility*.

See Database Configuration Type in *Creating WebLogic Domains Using the Configuration Wizard* .

Specifying JDBC Component Schema Information

Use the JDBC Component Schema screen to verify or specify details about the database schemas.

Verify that the values populated on the JDBC Component Schema screen are correct for all schemas. If you selected **RCU Data** on the previous screen, the schema table should already be populated appropriately. If you selected **Manual configuration** on the Database Configuration screen, you must configure the schemas listed in the table manually, before you proceed.

For an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), specify the connection credentials using only the **Connection URL String** option, and enter the connect string in the following format:

```
jdbc:oracle:thin:@TNS_alias?TNS_ADMIN=<path of the wallet files,
ojdbc.properties, and tnsnames.ora>
```

In the connect string, you must pass `TNS_alias` as the database name found in `tnsnames.ora`, and `TNS_ADMIN` property to the location of the wallet files, `ojdbc.properties`, and `tnsnames.ora`

Example connect string for Oracle Autonomous Transaction Processing-Dedicated (ATP-D) database:

```
jdbc:oracle:thin:@dbname_medium?TNS_ADMIN=/users/test/wallet_dbname/
```

Example connect string for Oracle Autonomous Transaction Processing-Shared (ATP-S) database:

```
jdbc:oracle:thin:@dbname_tp?TNS_ADMIN=/users/test/wallet_dbname/
```

For high availability environments, see the following sections in *High Availability Guide* for additional information on configuring data sources for Oracle RAC databases:

- Configuring Active GridLink Data Sources with Oracle RAC
- Configuring Multi Data Sources

See JDBC Component Schema in *Creating WebLogic Domains Using the Configuration Wizard* for more details about this screen.

Testing the JDBC Connections

Use the JDBC Component Schema Test screen to test the data source connections.

A green check mark in the Status column indicates a successful test. If you encounter any issues, see the error message in the Connection Result Log section of the screen, fix the problem, then try to test the connection again.

By default, the schema password for each schema component is the password you specified while creating your schemas. If you want different passwords for different schema components, manually edit them in the previous screen (JDBC Component Schema) by entering the password you want in the **Schema Password** column, against each row. After specifying the passwords, select the check box corresponding to the schemas that you changed the password in and test the connection again.

For more information about this screen, see JDBC Component Schema Test in *Creating WebLogic Domains Using the Configuration Wizard*.

Entering Credentials

Use the Credentials screen to set credentials for each key in the domain.

The following table lists the key names, and the values that you must specify for their respective username and password.

> **Note:**
>
> Ensure that you specify `keystore` as the username for the key **Keystore**, and `xelsysadm` as the username for the key **sysadmin**.

Table 4-5 Values to be Specified on the Credentials Screen

Key Name	Username	Password	Store Name
Keystore	`keystore`	Specify the password for keystore.	`oim`

Table 4-5 (Cont.) Values to be Specified on the Credentials Screen

Key Name	Username	Password	Store Name
OIMSchemaPassword	Specify the schema username for OIM operations database.	Specify the schema password of the OIM operations database schema owner.	`oim`
sysadmin	`xelsysadm`	Specify the sysadmin password.	`oim`
WebLogicAdminKey	Specify the username of the WebLogic administrator account for OIM domain.	Specify the password of the WebLogic administrator account for OIM domain.	`oim`

Specifying the Path to the Keystore Certificate or Key

Use the Keystore screen to specify either the path to the trusted certificate for each keystore, or the path to each keystore's private key and other private key information.

When you click in the Trusted Certificate, Private Key, or Identity Certificate fields, a browse icon appears to the right of the field. Click this icon to browse to the appropriate file.

For more information about this screen, see Keystore in *Creating WebLogic Domains Using the Configuration Wizard* .

Selecting Advanced Configuration

Use the Advanced Configuration screen to complete the domain configuration.

On the Advanced Configuration screen, select:

- Administration Server

  Required to properly configure the listen address of the Administration Server.
- Node Manager

  Required to configure Node Manager.
- Topology

  Required to configure the Oracle Identity Governance Managed Server.

Optionally, select other available options as required for your desired installation environment. The steps in this guide describe a standard installation topology, but you may choose to follow a different path. If your installation requirements extend to additional options outside the scope of this guide, you may be presented with additional screens to configure those options. For information about all Configuration Wizard screens, see Configuration Wizard Screens in *Creating WebLogic Domains Using the Configuration Wizard*.

Configuring the Administration Server Listen Address

Use the Administration Server screen to select the IP address of the host.

Select the drop-down list next to **Listen Address** and select the IP address of the host where the Administration Server will reside, or use the system name or DNS name that maps to a single IP address. Do *not* use `All Local Addresses`.

Do *not* specify any server groups for the Administration Server.

Configuring Node Manager

Use the Node Manager screen to select the type of Node Manager you want to configure, along with the Node Manager credentials.

Select **Per Domain Default Location** as the Node Manager type, then specify Node Manager credentials.

For more information about this screen, see Node Manager in *Creating WebLogic Domains Using the Configuration Wizard*.

For more about Node Manager types, see Node Manager Overview in *Administering Node Manager for Oracle WebLogic Server*.

Configuring Managed Servers for Oracle Identity Manager

On the Managed Servers screen, the new Managed Server named `oim_server1` and `soa_server1` are automatically created by default.

To configure Managed Servers for Oracle Identity Governance and Oracle SOA Suite:

1. In the Listen Address drop-down list, select the IP address of the host on which the Managed Server will reside or use the system name or DNS name that maps to a single IP address. Do not use `All Local Addresses`.
2. In the Server Groups drop-down list, make sure that `oim_server1` is associated with `OIM-MGD-SVRS` group and `soa_server1` is associated with `SOA-MGD-SVRS` group. This ensures that the correct service(s) target the Managed Servers you are creating.

   Server groups target Fusion Middleware applications and services to one or more servers by mapping defined application service groups to each defined server group. A given application service group may be mapped to multiple server groups if needed. Any application services that are mapped to a given server group are automatically targeted to all servers that are assigned to that group. For more information, see Application Service Groups, Server Groups, and Application Service Mappings in *Domain Template Reference*.
3. Click **Clone** to create a second Managed Server `oim_server2` of type `oim_server1`. Repeat it to create a second Managed Server `soa_server2` of type `soa_server1`.

   Configuring a second Managed Server is one of the steps needed to configure the standard topology for high availability. If you are not creating a highly available environment, then this step is optional.

   For more information about the high availability standard topology, see Understanding the Fusion Middleware Standard HA Topology in *High Availability Guide*.

   For more information about the next steps to prepare for high availability after your domain is configured, see Preparing Your Environment for High Availability.

These server names are referenced throughout this document; if you choose different names be sure to replace them as needed.

> **Tip:**
>
> For details about options on this screen, see Managed Servers in *Creating WebLogic Domains Using the Configuration Wizard*.

Configuring a Cluster for Oracle Identity Manager

Use the Clusters screen to create a new cluster. This is required for an Oracle Identity Governance high availability setup.

On the Clusters screen:

1. Click **Add**.
2. Specify `oim_cluster_1` in the Cluster Name field.
3. For the Cluster Address field, specify the *`ipaddress/hostname`*`:port`. For example:

   `ip_address_machine1``:`*`portnumber`*`,`*`ip_address_machine2`*`:`*`portnumber`*
4. Repeat the steps to add `soa_cluster1`.

By default, server instances in a cluster communicate with one another using unicast. If you want to change your cluster communications to use multicast, see Considerations for Choosing Unicast or Multicast in *Administering Clusters for Oracle WebLogic Server*.

You can also create clusters using Fusion Middleware Control. In this case, you can configure cluster communication (unicast or multicast) when you create the new cluster. See Create and configure clusters in *Oracle WebLogic Server Administration Console Online Help*.

> **Tip:**
>
> For more information about this screen, see Clusters in *Creating WebLogic Domains Using the Configuration Wizard*.

Defining Server Templates

If you are creating dynamic clusters for a high availability setup, use the Server Templates screen to define one or more server templates for domain.

To continue configuring the domain, click **Next**.

For steps to create a dynamic cluster for a high availability setup, see Using Dynamic Clusters in *High Availability Guide*.

Configuring Dynamic Servers

If you are creating dynamic clusters for a high availability setup, use the Dynamic Servers screen to configure the dynamic servers.

If you are *not* configuring a dynamic cluster, click **Next** to continue configuring the domain.

> **Note:**
>
> When you create dynamic clusters, keep in mind that after you assign the **Machine Name Match Expression**, you do not need to create machines for your dynamic cluster.

To create a dynamic cluster for a high availability setup, see Using Dynamic Clusters in *High Availability Guide*.

Assigning Oracle Identity Manager Managed Servers to the Cluster

If you are configuring a single-node non-clustered setup, click **Next** and go to next screen. Use the Assign Servers to Clusters screen to assign Managed Servers to a new *configured cluster*. A configured cluster is a cluster you configure manually. You do not use this screen if you are configuring a *dynamic cluster*, a cluster that contains one or more generated server instances that are based on a server template.

For more on configured cluster and dynamic cluster terms, see About Dynamic Clusters in *Understanding Oracle WebLogic Server*.

On the Assign Servers to Clusters screen:

1. In the Clusters pane, select the cluster to which you want to assign the Managed Servers; in this case, `oim_cluster1`.
2. In the Servers pane, assign `oim_server1` to `oim_cluster1` by doing one of the following:
   - Click once on `oim_server1` to select it, then click the right arrow to move it beneath the selected cluster (`oim_cluster1`) in the Clusters pane.
   - Double-click on `oim_server1` to move it beneath the selected cluster (`oim_cluster1`) in the Clusters pane.
3. Repeat to assign `soa_server1` to `soa_cluster1`.

> **Tip:**
>
> For more information about this screen, see Assign Servers to Clusters in *Creating WebLogic Domains Using the Configuration Wizard*.

Configuring Coherence Clusters

Use the Coherence Clusters screen to configure the Coherence cluster.

Leave the default port number as the Coherence cluster listen port. After configuration, the Coherence cluster is automatically added to the domain.

> **Note:**
>
> Setting the unicast listen port to `0` creates an offset for the Managed Server port numbers. The offset is `5000`, meaning the maximum allowed value that you can assign to a Managed Server port number is `60535`, instead of `65535`.

See Table 5-2 for more information and next steps for configuring Coherence.

For Coherence licensing information, see Oracle Coherence Products in *Licensing Information*.

Creating a New Oracle Identity Manager Machine

Use the Machines screen to create new machines in the domain. A machine is required so that Node Manager can start and stop servers.

> **Tip:**
>
> If you plan to create a high availability environment and know the list of machines your target topology requires, you can follow the instructions in this section to create all the machines at this time. For more about scale out steps, see Optional Scale Out Procedure in *High Availability Guide*.

To create a new Oracle Identity Governance machine so that Node Manager can start and stop servers:

1. Select the Machine tab (for Windows) or the UNIX Machine tab (for UNIX), then click **Add** to create a new machine.
2. In the Name field, specify a machine name, such as `oim_machine1`.
3. In the Node Manager Listen Address field, select the IP address of the machine in which the Managed Servers are being configured. You can also specify the host name for this field.

   You must select a specific interface and not `localhost`. This allows Coherence cluster addresses to be dynamically calculated.
4. Verify the port in the Node Manager Listen Port field.
5. Repeat these steps to add more machines, if required.

> **Note:**
>
> If you are extending an existing domain, you can assign servers to any existing machine. It is not necessary to create a new machine unless your situation requires it.

> **Tip:**
>
> For more information about this screen, see Machines in *Creating WebLogic Domains Using the Configuration Wizard*.

Assigning Servers to Oracle Identity Manager Machines

Use the Assign Servers to Machines screen to assign the Administration Server and Managed Servers to the new machine you just created.

On the Assign Servers to Machines screen:

1. In the Machines pane, select the machine to which you want to assign the servers; in this case, `oim_machine_1`.
2. In the Servers pane, assign `AdminServer` to `oim_machine_1` by doing one of the following:
   - Click once on `AdminServer` to select it, then click the right arrow to move it beneath the selected machine (`oim_machine_1`) in the Machines pane.
   - Double-click on `AdminServer` to move it beneath the selected machine (`oim_machine_1`) in the Machines pane.
3. Repeat these steps to assign all Managed Servers to their respective machines.

> **Tip:**
>
> For more information about this screen, see Assign Servers to Machines in *Creating WebLogic Domains Using the Configuration Wizard*.

Virtual Targets

If you have a WebLogic Server Multitenant (MT) environment, you use the Virtual Targets screen to add or delete virtual targets. For this installation (not a WebLogic Server MT environment), you do not enter any values; just select **Next**.

For details about this screen, see Virtual Targets in *Creating WebLogic Domains Using the Configuration Wizard*.

> **Note:**
>
> WebLogic Server Multitenant virtual targets are deprecated in WebLogic Server *12.2.1.4.0* and will be removed in the next release.

Partitions

The Partitions screen is used to configure partitions for virtual targets in WebLogic Server Multitenant (MT) environments. Select **Next** without selecting any options.

For details about options on this screen, see Partitions in *Creating WebLogic Domains Using the Configuration Wizard*.

> **Note:**
>
> WebLogic Server Multitenant domain partitions are deprecated in WebLogic Server *12.2.1.4.0* and will be removed in the next release.

Configuring Domain Frontend Host

The Domain Frontend Host screen can be used to configure the frontend host for the domain.

Select **Plain** or **SSL** and specify the respective host value.

Click **Next**.

Targeting the Deployments

The Deployments Targeting screen can be used to target the available deployments to the servers.

Make the required modifications, and click **Next**.

Targeting the Services

The Services Targeting screen can be used to target the available services to the Servers.

Make necessary modifications, and click **Next**.

File Stores

The File Stores screen lists the available file stores.

You can specify the Synchronous Write Policy for each of the file stores. After you make the changes, click **Next**.

Reviewing Your Configuration Specifications and Configuring the Domain

The Configuration Summary screen shows detailed configuration information for the domain you are about to create.

Review each item on the screen and verify that the information is correct. To make any changes, go back to a screen by clicking the **Back** button or selecting the screen in the navigation pane. Domain creation does not start until you click **Create**.

For more details about options on this screen, see Configuration Summary in *Creating WebLogic Domains Using the Configuration Wizard*.

Writing Down Your Domain Home and Administration Server URL

The End of Configuration screen shows information about the domain you just configured.

Make a note of the following items because you need them later:

- Domain Location

- Administration Server URL

You need the domain location to access scripts that start Node Manager and Administration Server, and you need the URL to access the Administration Server.

Click **Finish** to dismiss the Configuration Wizard.

Additional Domain Configuration

Use the Configuration Wizard to update the domain.

For information on other methods to create domains, see Additional Tools for Creating, Extending, and Managing WebLogic Domains in *Creating WebLogic Domains Using the Configuration Wizard*.

Complete the following steps:

1. Change to the following directory:
   (UNIX) *ORACLE_HOME*/oracle_common/common/bin

   (Windows) *ORACLE_HOME*\oracle_common\common\bin

   Where, *ORACLE_HOME* is your 12c (12.2.1.4.0) Oracle home.
2. Enter the following command:
   (UNIX) ./config.sh

   (Windows) config.cmd

   The configuration screen is displayed.
3. On the Configuration Type screen, select **Update an existing domain**.
4. In the Domain Location field, specify the Domain home directory.
5. On the Templates screen, select **Update Domain Using Custom Template**.
6. In the Template location field, specify:
   ORACLE_HOME/soa/common/templates/wls/oracle.soa.classic.domain_template.jar
7. Complete the configuration wizard by entering the required values in the respective screens. For information about the configuration screens, see Navigating the Configuration Wizard Screens to Create and Configure the Domain.

Performing Post-Configuration Tasks

After you configure the Oracle Identity Governance domain, perform the necessary post-configuration tasks.

Topics

- Running the Offline Configuration Command
  After you configure the Oracle Identity Governance domain, run the offlineConfigManager script to perform post configuration tasks.

Running the Offline Configuration Command

After you configure the Oracle Identity Governance domain, run the `offlineConfigManager` script to perform post configuration tasks.

Ensure that you run this command before you start any server. To run the `offlineConfigManager` command, do the following:

1. Set the following environment variables to the right values:
   - `DOMAIN_HOME`
   - `JAVA_HOME`
2. Ensure that you have execute permissions for the file *`OIM_HOME`*`/server/bin/offlineConfigManager.sh`.
3. Run the following command from the location *`OIM_HOME`*`/server/bin/`:
   - On Unix: `./offlineConfigManager.sh`
   - On Windows: `offlineConfigManager.bat`

> **Note:**
>
> *`OIM_HOME`* refers to *`ORACLE_HOME`*`/idm`.

Starting the Servers

After a successful configuration, start all processes and servers, including the Administration Server and any Managed Servers.

The components may be dependent on each other so they must be started in the correct order.

> **Note:**
>
> The procedures in this section describe how to start servers and process using the WLST command line or a script. You can also use the Oracle Fusion Middleware Control and the Oracle WebLogic Server Administration Console. See Starting and Stopping Administration and Managed Servers and Node Manager in *Administering Oracle Fusion Middleware.*

To start your Fusion Middleware environment, follow the steps below.

Step 1: Start Node Manager

To start Node Manager, use the `startNodeManager` script:

- (UNIX) *`EXISTING_DOMAIN_HOME`*`/bin/startNodeManager.sh`
- (Windows) *`EXISTING_DOMAIN_HOME`*`\bin\startNodeManager.cmd`

Step 2: Start the Administration Server

When you start the Administration Server, you also start the processes running in the Administration Server, including the WebLogic Server Administration Console and Fusion Middleware Control.

To start the Administration Server, use the `startWebLogic` script:

- (UNIX) *EXISTING_DOMAIN_HOME*`/bin/startWebLogic.sh`
- (Windows) *EXISTING_DOMAIN_HOME*`\bin\startWebLogic.cmd`

When you created the domain, if you selected **Production Mode** on the **Domain Mode and JDK** screen, a prompt for the Administrator user login credentials is displayed. Provide the same credentials that you provided on the **Administrator Account** screen.

For an Autonomous Transaction Processing database (both Oracle Autonomous Transaction Processing-Dedicated (ATP-D) and Oracle Autonomous Transaction Processing-Shared (ATP-S)), when you access the Sysadmin Console (`http://`*<machine_name>*`:`*<oim_server_port>*`/sysadmin`) and the OIM Console (`http://`*<machine_name>*`:`*<oim_server_port>*`/identity`), JET UI does not work and blank pages are displayed, and the following error message may be displayed in the Administration Server logs.

Example message:

```
<AdminServer> <[ACTIVE] ExecuteThread: '63' for queue:
'weblogic.kernel.Default (self-tuning)'> <weblogic> <>
<16023522-e47f-40f4-a66f-7ea3729188d1-00000064> <1628079696204>
<[severity-value: 8] [rid: 0] [partition-id: 0] [partition-name: DOMAIN] >
<BEA-240003> <Administration Console encountered the following error:
java.lang.NoSuchMethodError:
org.glassfish.jersey.internal.LocalizationMessages.WARNING_PROPERTIES()Ljava/
l ang/String; at
org.glassfish.jersey.internal.config.SystemPropertiesConfigurationModel.getPr
operties(SystemPropertiesConfigurationModel.java:122) at
org.glassfish.jersey.internal.config.SystemPropertiesConfigurationProvider.ge
tProperties(SystemPropertiesConfigurationProvider.java:29) at
org.glassfish.jersey.internal.config.ExternalPropertiesConfigurationFactory.r
eadExternalPropertiesMap(ExternalPropertiesConfigurationFactory.java:55) at
org.glassfish.jersey.internal.config.ExternalPropertiesConfigurationFactory.c
onfigure(ExternalPropertiesConfigurationFactory.java:72) at
org.glassfish.jersey.internal.config.ExternalPropertiesConfigurationFeature.c
onfigure(ExternalPropertiesConfigurationFeature.java:26) at
org.glassfish.jersey.model.internal.CommonConfig.configureFeatures(CommonConf
ig.java:730)
```

> **Note:**
>
> If JET UI does not work, blank pages are displayed for the following screens:
>
> - OIM Console
>   - Application onboarding (AOB)
>   - Account > Resource History
>   - Open Tasks
> - Sysadmin Console
>   - IT Resource Create/Search
>   - Manage Connector
>   - Import/Export (Deployment Manager)

The workaround is to restart the servers, Administration Server, Oracle SOA Server, and Oracle Identity Manager (OIM) Server from the terminal after unsetting classpath using the command:

```
EXPORT CLASSPATH=
```

> **Note:**
>
> You must restart the servers in the following order:
>
> - Administration Server
> - Oracle SOA Server
> - Oracle OIM Server

Step 3: Start the Managed Servers

- If Node Manager is *not* configured, start the Managed Servers using the following instructions:
  Start the Oracle SOA Suite Managed Server first and then the Oracle Identity Governance Managed Server.

  To start a WebLogic Server Managed Server, use the `startManagedWebLogic` script:

  - (UNIX) *`EXISTING_DOMAIN_HOME`*`/bin/startManagedWebLogic.sh` *`managed_server_name admin_url`*
  - (Windows) *`EXISTING_DOMAIN_HOME`*`\bin\startManagedWebLogic.cmd` *`managed_server_name admin_url`*

  When prompted, enter your user name and password. This is the same user name and password which you provided in administrator account screen when creating the domain.

> **Note:**
>
> The startup of a Managed Server will typically start the applications that are deployed to it. Therefore, it should not be necessary to manually start applications after the Managed Server startup.

- If Node Manager is configured, start the Managed Servers using the following instructions:

  1. Launch the Administration Console:

     a. Using a web browser, open the following URL:

        ```
        http://hostname:port/console
        ```

        Where:

        - *hostname* is the administration server host.
        - *port* is the administration server port on which the host server is listening for requests (7001 by default)

     b. When the login page appears, enter the user name and password you used to start the Administration Server.

  2. Start Managed Servers from the Administration Console. For instructions, see Start Managed Servers from the Administration Console.

Integrating Oracle Identity Governance with Oracle SOA Suite

If you wish to integrate Oracle Identity Governance with Oracle SOA Suite, use the Enterprise Manager console to do the same.

To integrate Oracle Identity Governance with Oracle SOA Suite, do the following:

1. Log in to Oracle Fusion Middleware Control:

   ```
   http://administration_server_host:administration_server_port/em
   ```

   The Administration Server host and port number were in the URL on the End of Configuration screen (Writing Down Your Domain Home and Administration Server URL). The default Administration Server port number is `7001`.

   The login credentials were provided on the Administrator Account screen (Configuring the Administrator Account).

2. Click **weblogic_domain** and then click **System Mbean Browser**.
3. In the search box, enter **OIMSOAIntegrationMBean**, and click Search. The mbean is displayed.

> **Note:**
>
> If Oracle Identity Governance is still starting (coming up) or is just started (RUNNING MODE), the Enterprise Manager does not show any Mbeans defined by OIG. Wait for two minutes for the server to start, and then try searching for the Mbean in **System Mbean Browser** of the Enterprise Manager,.

4. Go to the **Operations** tab of mbean, and select **integrateWithSOAServer**.
5. Enter the following required attributes:
   - **Weblogic Administrator User name**: `Weblogic Administrator User name`
   - **Weblogic Administrator User Password**: `The password for the WebLogic administrator account`
   - **OIM Front end URL**: `http://<HOSTNAME>:<OIM_server_port>`
   - **OIM External Front end URL**: `http://<HOSTNAME>:<OIM_server_port>`
   - **SOA SOAP URL**: `http://<HOSTNAME>:<SOA_server_port>`
   - **SOA RMI URL**: `t3://<HOSTNAME>:<SOA_server_port>`
   - **UMS Webservice URL**: `http://<HOSTNAME>:<SOA_server_port>/ucs/messaging/webservice`

> **Note:**
>
> When there is a load balancer, the above value differs:
>
> - If `OIM >= 11.1.2.2.0` then select **OIM Front end URL**.
> - If `OIM < 11.1.2.2.0` then select **OIM External Front end URL**.
> - **SOA RMI URL:** `t3://<soahost1>:<soalistenport1>,<soahost2>:<soalistenport2>`
>
> The **SOA SOAP URL**, **SOA RMI URL**, and **UMS Webservice URL** attributes in Oracle Identity Governance with Oracle SOA Suite can be seen on the EM console only if you have applied the OPatch 28186730 and the OIM bundle patch 12.2.1.4.210428 (p32829648_122140_Generic.zip) on the 12.2.1.4.0 ORACLE_HOME.

6. Click **Invoke**.

Verifying the Configuration

After completing all configuration steps, you can perform additional steps to verify that your domain is properly configured.

By using a Web browser, go to the URL: `http://HOSTNAME:PORT/identity`

In this URL, *`HOSTNAME`* represents the name of the computer hosting the application server and *`PORT`* refers to the port on which the Oracle Identity Governance server is listening.

For information about integrating Oracle Identity Governance with other Identity Management components, see Introduction to IdM Suite Components Integration in *Integration Guide for Oracle Identity Management Suite*.

For more information about performing additional domain configuration tasks, see Performing Additional Domain Configuration Tasks.

Analyzing the Bootstrap Report

When you start the Oracle Identity Governance server, the bootstrap report is generated at `DOMAIN_HOME/servers/oim_server1/logs/BootStrapReportPreStart.html`.

The bootstrap report `BootStrapReportPreStart.html` is an html file that contains information about the topology that you have deployed, the system level details, the connection details like the URLs to be used, the connectivity check, and the task execution details. You can use this report to check if the system is up, and also to troubleshoot the issues, post-configuration.
Every time you start the Oracle Identity Governance server, the bootstrap report is updated.

Sections in the Bootstrap Report

- **Topology Details**

  This section contains information about your deployment. It shows whether you have configured a cluster setup, SSL enabled, or upgraded an Oracle Identity Manager environment from 12c (12.2.1.3.0) to 12c (12.2.1.4.0).

- **System Level Details**

  This section contains information about the JDK version, Database version, JAVA_HOME, DOMAIN_HOME, OIM_HOME, and ORACLE_HOME.

- **Connection Details**

  This section contains information about the connect details like the Administration URL, OIM Front End URL, SOA URL, and RMI URL.

  This also shows whether the Administration Server, Database, and SOA server is up or not.

- **Execution Details**

  This section lists the various tasks and their statuses.

Installing and Accessing the Oracle Identity Governance Design Console

If you wish to set up only the Oracle Identity Governance Design Console in a machine where OIG server is not configured, then you must install Oracle Identity Governance 12c (12.2.1.4.0) in standalone mode, and then invoke the Design Console.

To install the Oracle Identity Governance Design Console, do the following:

1. Start the installation program by running the `java` executable from the JDK directory.

> **Note:**
>
> No prerequisite software is required to install Oracle Identity Governance Design Console.

For example:

- (UNIX) `/home/Oracle/Java/jdk1.8.0_211/bin/java -jar fmw_12.2.1.4.0_idm.jar`
- (Windows) `C:\home\Oracle\Java\jdk1.8.0_211\bin\java -jar fmw_12.2.1.4.0_idm.jar`

2. The installer shows a series of screens where you verify or enter information.. The following table lists the order in which installer screens appear. If you need additional help with an installation screen, click **Help**.

Table 4-6 Install Screens

Screen	Description
Installation Inventory Setup	On Linux or UNIX operating systems, this screen opens if this is the first time you are installing any Oracle product on this host. Specify the location where you want to create your central inventory. Make sure that the operating system group name selected on this screen has write permissions to the central inventory location. See About the Oracle Central Inventory in *Installing Software with the Oracle Universal Installer*. This screen does not appear on Windows operating systems.
Welcome	Review the information to make sure that you have met all the prerequisites, then click **Next**.
Auto Updates	Select to skip automatic updates, select patches, or search for the latest software updates, including important security updates, through your My Oracle Support account.
Installation Location	Specify your Oracle home directory location. You can click **View** to verify and ensure that you are installing in the correct Oracle home. **Note:** Ensure that the Oracle Home path does not contain space.
Installation Type	Use the **Standalone** Installation Type. Standalone mode is a type of installation that is managed independently of WebLogic Server. The only component that you can install using standalone mode is the Oracle Identity Governance Design Console.
Prerequisite Checks	This screen verifies that your system meets the minimum necessary requirements. To view the list of tasks that gets verified, select **View Successful Tasks**. To view log details, select **View Log**. If any prerequisite check fails, then an error message appears at the bottom of the screen. Fix the error and click **Rerun** to try again. To ignore the error or the warning message and continue with the installation, click **Skip** (not recommended).

Table 4-6 (Cont.) Install Screens

Screen	Description
Installation Summary	Use this screen to verify installation options you selected. If you want to save these options to a response file, click **Save Response File** and enter the response file location and name. The response file collects and stores all the information that you have entered, and enables you to perform a silent installation (from the command line) at a later time. Click **Install** to begin the installation.
Installation Progress	This screen shows the installation progress. When the progress bar reaches 100% complete, click **Finish** to dismiss the installer, or click **Next** to see a summary.
Installation Complete	This screen displays the Installation Location and the Feature Sets that are installed. Review this information and click **Finish** to close the installer.

To access the Oracle Identity Governance Design Console, do the following:

1. Ensure that the `JAVA_HOME` environment variable is set to the location of the certified JDK. For example:
   - (UNIX) `setenv JAVA_HOME /home/Oracle/Java/jdk1.8.0_211`
   - (Windows) `set JAVA_HOME=C:\home\Oracle\Java\jdk1.8.0_211`
2. Invoke the Design Console by running the following command from the location *`ORACLE_HOME`*`\idm\designconsole`:
   - (UNIX) `sh xlclient.sh`
   - (Windows) `xlclient.cmd`

   Enter the following details when prompted:
   - `Server url`: Enter the Oracle Identity Governance server URL in the format `t3://`*`oim_server_hostname`*`:`*`oimport`*.
   - `User ID`: Enter the OIG Administrator user login. For example, `xelsysadm`.
   - `Password`: Enter the OIG Administrator user password. For example, `xelsysadm_password`.

Troubleshooting

This section lists the common issues encountered while configuring Oracle Identity Governance and their workarounds.

Topics

- Description of the Log Codes
  When you encounter any error during the Oracle Identity Governance 12c (12.2.1.4.0) installation, search for the log code in the *`DOMAIN_HOME`*`/servers/`*`oim_server`*`/logs/oim-diagnostic.log` file to diagnose the issue.
- Exception in the Oracle Identity Manager Server Logs After Starting the Servers
  After you configure the Oracle Identity Manager domain, when you start the servers, `"Unable to resolve 'TaskQueryService'"` exception is seen in the Oracle Identity Manager (OIM) Server logs, which can be ignored.

- Oracle Identity Manager Bootstrap Fails with Hostname Verification Error
  If the Oracle Identity Manager bootstrap fails with the following SSL hostname verification failing error, use the workaround described in this section:
- Error When Accessing Pending Approvals Page in a Multinode Setup
  In a Oracle Identity Governance multinode setup, the following error is displayed when you access the Pending Approvals page on a remote node:
- OIM Gridlink Datasources Show Suspended State When 11.2.0.4.0 RAC Database is Used
  When you run the Configuration Wizard to configure Oracle Identity Manager gridlink datasources with 11.2.0.4.0 RAC Database, the following warning is displayed:
- Server Consoles are Inaccessible in a Clustered Domain
  After you configure the Oracle Identity Governance domain, the Administration Server console and the managed Server consoles are inaccessible.
- OIM Server Fails to Come up Due to SOA Server not Completely Up
  If the Oracle SOA Server (SOA) is not up completely, the Oracle Identity Manager (OIM) Server fails to start.
- Oracle Identity Manager Server Throws OutOfMemoryError
  After you configure Oracle Identity Manager 12c (12.2.1.4.0), when you start the OIM 12c (12.2.1.4.0) Server, `OutOfMemoryError` is thrown.
- 'ADFContext leak detected' Message in the OIM Server Logs
  When you start the Oracle Identity Manager (OIM) 12c (12.2.1.4.0) server, the following error is seen in the OIM server logs:
- ADF Controller Exception in the SOA Server Logs
  After you configure Oracle Identity Governance 12c (12.2.1.4.0), when you start the Oracle SOA Suite (SOA) server, the following exception is shown in the SOA server logs:

Description of the Log Codes

When you encounter any error during the Oracle Identity Governance 12c (12.2.1.4.0) installation, search for the log code in the *DOMAIN_HOME*`/servers/`*oim_server*`/logs/oim-diagnostic.log` file to diagnose the issue.

The following are log codes and their descriptions for various tasks:

- IAM-3070001 — Error loading configuration required for Bootstrap
- IAM-3070002 — Could not connect to DB using CSF Credentials, Please verify crednetials seeded in CSF under key
- IAM-3070003 — Could not connect to WLS using CSF credentials ,Please verify credentials seeded in CSF for
- IAM-3070004 — Validation for CSF Credentials failed. Exiting OIM_CONFIG, Please verify and fix CSF Credentials
- IAM-3070005 — Validation for CSF Credentials Successful
- IAM-3070006 — Task Not Found
- IAM-3070007 — Task failed
- IAM-3070008 — BootStrap configuration Failed
- IAM-3070009 — BootStrap configuration Successful

- IAM-3070010 — Successfully completed

Exception in the Oracle Identity Manager Server Logs After Starting the Servers

After you configure the Oracle Identity Manager domain, when you start the servers, `"Unable to resolve 'TaskQueryService'"` exception is seen in the Oracle Identity Manager (OIM) Server logs, which can be ignored.

The following exception is displayed in the OIM Server logs:

```
javax.naming.NameNotFoundException: Unable to resolve 'TaskQueryService'.
Resolved ''; remaining name 'TaskQueryService'
```

This exception can be ignored.

Oracle Identity Manager Bootstrap Fails with Hostname Verification Error

If the Oracle Identity Manager bootstrap fails with the following SSL hostname verification failing error, use the workaround described in this section:

```
<Warning> <Security> <BEA-090960> <The servers
SSL configuration is not available. There will potentially be SSL handshake
failures.>
<Nov 28, 2018 9:04:32 AM PDT> <Warning> <Security> <BEA-090924> <JSSE has
been selected by default, since the SSLMBean is not available.>
<Nov 28, 2018 9:04:32 AM PDT> <Info> <Security> <BEA-090908> <Using the
default WebLogic SSL Hostname Verifier implementation.>
<Nov 28, 2018 9:04:34 AM PDT> <Notice> <Security> <BEA-090169> <Loading
trusted certificates from the kss keystore file kss://system/trust.>
Nov 28, 2018 9:04:34 AM
oacle.security.opss.internal.runtime.ServiceContextManagerImpl getContext
WARNING: Bootstrap services are used by OPSS internally and clients should
never need to directly read/write bootstrap credentials. If required, use
Wlst or configuration management interfaces.
<Nov 28, 2018 9:04:34 AM PDT> <Notice> <Security> <BEA-090169> <Loading
trusted certificates from the jks keystore file
/host/jdk1.8.0_171/jre/lib/security/cacerts.>
<Nov 28, 2018 9:04:34 AM PDT> <Info> <Management> <BEA-141307> <Unable to
connect to the Administration Server. Waiting 5 second(s) to retry (attempt
number 1 of 3).>
```

To resolve this issue, start the Oracle Identity Governance Managed Server using the following command:

- On Unix:

  `./startManagedWebLogic.sh` *`oim_server_name`* `t3://`*`admin_server_host`*`:`*`port`*

- On Windows:

  `startManagedWebLogic.cmd` *`oim_server_name`* `t3://`*`admin_server_host`*`:`*`port`*

In this command, you must specify the non-SSL port for *`port`*.

Error When Accessing Pending Approvals Page in a Multinode Setup

In a Oracle Identity Governance multinode setup, the following error is displayed when you access the Pending Approvals page on a remote node:

```
[oim_server1] [ERROR] [] [oracle.iam] [tid:
[ACTIVE].ExecuteThread: '0' for queue: 'weblogic.kernel.Default
(self-tuning)'] [userId: xelsysadm] [ecid:
cea9a502-afb8-4d3d-85a4-cb61d2878065-0000276e,0] [APP:
oracle.iam.console.identity.self-service.ear] [partition-name: DOMAIN]
[tenant-name: GLOBAL] [DSID: 0000LfRXW3_7Y7QLIag8yf1OmuCL000004]
Unable to
retrieve User View
Listoracle.bpel.services.workflow.client.WorkflowServiceClientException
:
javax.naming.CommunicationException: Failed to initialize JNDI
context, tried
2 time or times totally, the interval of each time is 0ms. [[
t3://host.example.com:1234: Destination 10.10.10.1, 1234
unreachable.; nested exception is:
        java.net.ConnectException: Connection refused; No available
router to
destination.; nested exception is:
        java.rmi.ConnectException: No available router to destination.
[Root
exception is java.net.ConnectException: t3://host.example.com:1234:
Destination 10.10.10.1, 1234 unreachable.; nested exception is:
        java.net.ConnectException: Connection refused; No available
router to
destination.; nested exception is:
        java.rmi.ConnectException: No available router to destination.]
```

To resolve this, you must use the machine name of the second node during the domain creation step, that is, when running the configuration wizard on the first node. After this, you must proceed with the pack and unpack command.

OIM Gridlink Datasources Show Suspended State When 11.2.0.4.0 RAC Database is Used

When you run the Configuration Wizard to configure Oracle Identity Manager gridlink datasources with 11.2.0.4.0 RAC Database, the following warning is displayed:

```
<Nov 28, 2017 2:45:44,157 AM MDT> <Warning> <JDBC> <BEA-001129>
<Received
exception while creating connection for pool
"ApplicationDB": Listener refused the connection with the following
error:
ORA-12516, TNS:listener could not find available handler with matching
protocol stack
```

The data source is pushed to suspended state if the connection fails in the retry after waiting for TEST Frequency. To resolve this, you must manually resume the suspended data sources by doing the following:

1. Navigate to the data source that you want to resume:
2. Go to the **Control** tab.
3. On the Control page, select the instances of the data source that you want to resume.

   Date source instances are listed by the server on which they are deployed.
4. Click **Resume** and then click **Yes** to confirm the action.

Results are displayed at the top of the page, and the state of the selected data source instances is changed to Running.

Server Consoles are Inaccessible in a Clustered Domain

After you configure the Oracle Identity Governance domain, the Administration Server console and the managed Server consoles are inaccessible.

To resolve this, either specify the IP address of machine as listen address for machines having multiple interfaces, or disable all other interfaces.
If you wish to enter machine name as listen address in a clustered or non-clustered domain, disable all other interfaces.

OIM Server Fails to Come up Due to SOA Server not Completely Up

If the Oracle SOA Server (SOA) is not up completely, the Oracle Identity Manager (OIM) Server fails to start.

The following error is displayed when OIM Server fails to start if the SOA Server is not completely up:

```
 Could not fetch ServerRuntime mbean for
soa_server1. Server seems to be down!
```

To resolve this, restart the OIM Server.

Oracle Identity Manager Server Throws OutOfMemoryError

After you configure Oracle Identity Manager 12c (12.2.1.4.0), when you start the OIM 12c (12.2.1.4.0) Server, `OutOfMemoryError` is thrown.

The following error is seen in the OIM server logs for this issue:

```
[oim_server1] [NOTIFICATION] []
[oracle.iam.oimdataproviders.impl] [tid: [ACTIVE].ExecuteThread: '9' for
queue: 'weblogic.kernel.Default (self-tuning)'] [userId: xelsysadm] [ecid:
5679ce10-f0df-457f-88f1-6bc04e10aa13-000013b1,0] [APP: oim-runtime]
[partition-name: DOMAIN] [tenant-name: GLOBAL] [DSID:
0000Lg0PPYTBd5I_Ipt1if1OpGGi00000U] RM_DEBUG_PERF - 2018-11-28 06:09:51.087
-
search criteria = arg1 = (usr_key) EQUAL arg2 = (1)[[
 query = Select usr.usr_key, usr.usr_status  from usr where usr.usr_key = ?
 time = 1
```

```
]]
[2018-11-28T06:09:52.286-07:00] [oim_server1] [NOTIFICATION] []
[oracle.iam.oimdataproviders.impl] [tid: [ACTIVE].ExecuteThread: '9'
for
queue: 'weblogic.kernel.Default (self-tuning)'] [userId: xelsysadm]
[ecid:
5679ce10-f0df-457f-88f1-6bc04e10aa13-000013b1,0] [APP: oim-runtime]
[partition-name: DOMAIN] [tenant-name: GLOBAL] [DSID:
0000Lg0PPYTBd5I_Ipt1if1OpGGi00000U]
oracle.iam.oimdataproviders.impl.OIMUserDataProvider
[2018-11-28T06:11:52.171-07:00] [oim_server1] [ERROR] [ADFC-50018]
[oracle.adfinternal.controller.application.AdfcExceptionHandler] [tid:
[ACTIVE].ExecuteThread: '27' for queue: 'weblogic.kernel.Default
(self-tuning)'] [userId: xelsysadm] [ecid:
5679ce10-f0df-457f-88f1-6bc04e10aa13-000013e0,0] [APP:
oracle.iam.console.identity.self-service.ear] [partition-name: DOMAIN]
[tenant-name: GLOBAL] [DSID: 0000Lg0RtM9Bd5I_Ipt1if1OpGGi00000V] ADFc:
No
exception handler was found for an application exception.[[
java.lang.OutOfMemoryError: GC overhead limit exceeded ]
```

To resolve this issue, do the following (on Linux):

1. Ensure that you set the following parameters in the `/etc/security/limits.conf` file, to the specified values:
   - `FUSION_USER_ACCOUNT soft nofile 32767`
   - `FUSION_USER_ACCOUNT hard nofile 327679`
2. Ensure that you set `UsePAM` to `Yes` in the /etc/ssh/sshd_config file.
3. Restart `sshd`.
4. Log out (or reboot) and log in to the system again.

Before you start the Oracle Identity Manager 12c (12.2.1.4.0) Server, run the following command to increase the limit of open files, so that you do not hit into memory issues:
`limit maxproc 16384`

‘ADFContext leak detected’ Message in the OIM Server Logs

When you start the Oracle Identity Manager (OIM) 12c (12.2.1.4.0) server, the following error is seen in the OIM server logs:

```
2b8fd3a0-06e3-4de6-be10-801551745664-000000a5,0] [partition-name:
DOMAIN]
[tenant-name: GLOBAL] ADFContext leak detected.[[
oracle.adf.share.ADFContext.setAsCurrent(ADFContext.java:1501)
oracle.adf.mbean.share.AdfMBeanInterceptor.resetADFIfNeeded(AdfMBeanInt
erceptor.java:140)
```

This has no impact on the functionality, and therefore you can ignore this error.

ADF Controller Exception in the SOA Server Logs

After you configure Oracle Identity Governance 12c (12.2.1.4.0), when you start the Oracle SOA Suite (SOA) server, the following exception is shown in the SOA server logs:

```
oracle.adf.controller.ControllerException: ADFC-12013: Controller state has
not been initialized for the current request.
```

This does not impact the functionality, and therefore it can be ignored.

5

Next Steps After Configuring the Domain

After you configure a product domain, there are additional tasks that you may want to perform.

- Performing Basic Administrative Tasks
  Review the administrative tasks you will likely want to perform on a new domain.
- Performing Additional Domain Configuration Tasks
  Review additional configuration tasks you will likely want to perform on a new domain.
- Preparing Your Environment for High Availability
  Scaling out for high availability requires additional steps.

Performing Basic Administrative Tasks

Review the administrative tasks you will likely want to perform on a new domain.

Table 5-1 Basic Administration Tasks for a New Domain

Task	Description	More Information
Getting familiar with Fusion Middleware administration tools	Get familiar with various tools that you can use to manage your environment.	See Overview of Oracle Fusion Middleware Administration Tools in *Administering Oracle Fusion Middleware*.
Starting and stopping products and servers	Learn how to start and stop Oracle Fusion Middleware, including the Administration Server, Managed Servers, and components.	See Starting and Stopping Oracle Fusion Middleware in *Administering Oracle Fusion Middleware*.
Configuring Secure Sockets Layer (SSL)	Learn how to set up secure communications between Oracle Fusion Middleware components using SSL.	See Configuring SSL in Oracle Fusion Middleware in *Administering Oracle Fusion Middleware*.
Monitoring Oracle Fusion Middleware	Learn how to keep track of the status of Oracle Fusion Middleware components.	See Monitoring Oracle Fusion Middleware in *Administering Oracle Fusion Middleware*.
Understanding Backup and Recovery Procedures	Learn the recommended backup and recovery procedures for Oracle Fusion Middleware.	See Introduction to Backup and Recovery in *Administering Oracle Fusion Middleware*.
Understanding Performance Tuning	Learn how to tune the Oracle Identity and Access Management Suite components to improve performance.	See Oracle Identity and Access Management Performance Tuning in *Tuning Performance*.

Performing Additional Domain Configuration Tasks

Review additional configuration tasks you will likely want to perform on a new domain.

Table 5-2 Additional Domain Configuration Tasks

Task	Description	More Information
Deploying Applications	Learn how to deploy your applications to Oracle Fusion Middleware.	See Deploying Applications in *Administering Oracle Fusion Middleware*.
Adding a Web Tier front-end to your domain	Oracle Web Tier hosts Web pages (static and dynamic), provides security and high performance along with built-in clustering, load balancing, and failover features. In particular, the Web Tier contains Oracle HTTP Server.	To install and configure Oracle HTTP Server in the WebLogic Server domain, see Configuring Oracle HTTP Server in a WebLogic Server Domain in *Installing and Configuring Oracle HTTP Server*.
Tuning and configuring Coherence for your topology	The standard installation topology includes a Coherence cluster that contains storage-enabled Managed Coherence Servers. This configuration is a good starting point for using Coherence, but depending upon your specific requirements, consider tuning and reconfiguring Coherence to improve performance in a production environment.	For more information about Coherence clusters, see Configuring and Managing Coherence Clusters in *Administering Clusters for Oracle WebLogic Server*. For information on tuning Coherence, see Performance Tuning in *Administering Oracle Coherence*. For information on storing HTTP session data in Coherence, see Using Coherence*Web with WebLogic Server in *Administering HTTP Session Management with Oracle Coherence*Web*. For more about creating and deploying Coherence applications, see Getting Started in *Developing Oracle Coherence Applications for Oracle WebLogic Server*.

Preparing Your Environment for High Availability

Scaling out for high availability requires additional steps.

Table 5-3 provides a list of tasks to perform if you want to scale out your standard installation environment for high availability.

Table 5-3 Tasks Required to Prepare Your Environment for High Availability

Task	Description	More Information
Scaling out to multiple host computers	To enable high availability, it is important to provide failover capabilities to another host computer. That way, if one computer goes down, your environment can continue to serve the consumers of your deployed applications.	See Scaling Out a Topology (Machine Scale Out) in the *High Availability Guide*.
Configuring high availability for your Web Tier components.	If you have added a Web tier front-end, then you must configure the Web Tier for high availability, as well as the WebLogic Server software.	See Configuring High Availability for Web Tier Components in *High Availability Guide*.

Table 5-3 (Cont.) Tasks Required to Prepare Your Environment for High Availability

Task	Description	More Information
Setting up a front-end load balancer	A load balancer can be used to distribute requests across servers more evenly.	See Server Load Balancing in a High Availability Environment and Configuring Load Balancer Virtual Server Names and Ports in *High Availability Guide*.
Configuring Node Manager	Node Manager enables you to start, shut down, and restart the Administration Server and Managed Server instances from a remote location. This document assumes you have configured a per-domain Node Manager. Review the Node Manager documentation, for information on advanced Node Manager configuration options and features.	See Advanced Node Manager Configuration in *Administering Node Manager for Oracle WebLogic Server*.

6

Configuring High Availability for Oracle Identity Governance Components

This chapter describes how to design and deploy a high availability environment for Oracle Identity Governance.

Oracle Identity Governance (OIG) is a user provisioning and administration solution that automates the process of adding, updating, and deleting user accounts from applications and directories. It also improves regulatory compliance by providing granular reports that attest to who has access to what. OIG is available as a stand-alone product or as part of Oracle Identity and Access Management Suite.

For details about OIG, see in Product Overview for Oracle Identity Governance in *Administering Oracle Identity Governance*.

> **Note:**
>
> Oracle Identity Governance and Oracle Identity Manager product name references in the documentation mean the same.

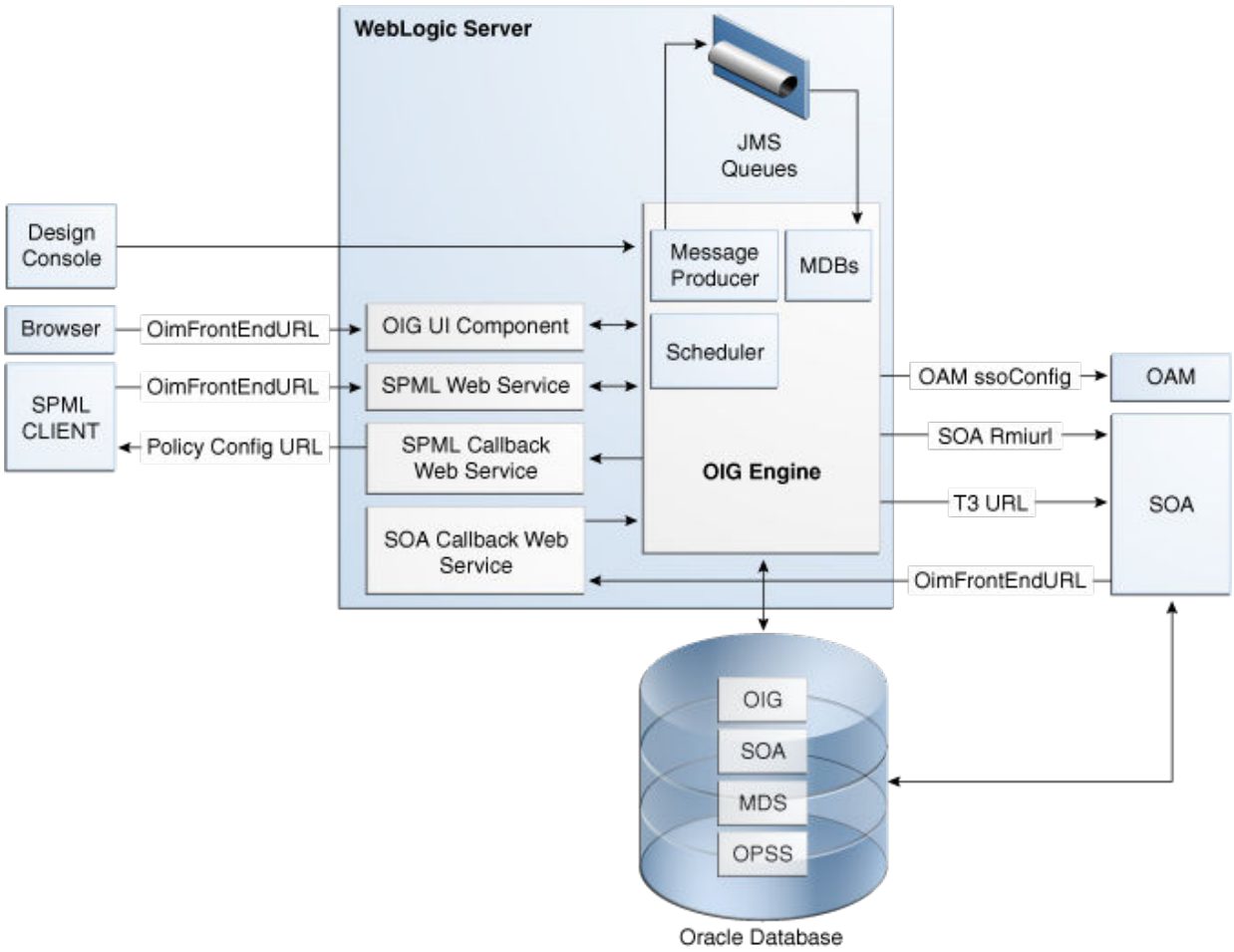
- Oracle Identity Governance Architecture
  Oracle Identity Governance architecture consists of its components, runtime processes, process lifecycle, configuration artifacts, external dependencies, and log files.
- Oracle Identity Governance High Availability Concepts
  The concepts related to Oracle Identity Governance High Availability are OIG high availability architecture, starting and stopping OIG cluster, and cluster-wide configuration changes.
- High Availability Directory Structure Prerequisites
  A high availability deployment requires product installations and files to reside in specific directories. A standard directory structure makes facilitates configuration across nodes and product integration.
- Oracle Identity Governance High Availability Configuration Steps
  Oracle Identity Governance high availability configuration involves setting the prerequisites, configuring the domain, post-installation steps, starting servers, SOA integration, validating managed server instances, and scaling up and scaling out Oracle Identity Governance.
- Preparing for Shared Storage
  Oracle Fusion Middleware allows you to configure multiple WebLogic Server domains from a single Oracle home. This allows you to install the Oracle home in a single location on a shared volume and reuse the Oracle home for multiple host installations.
- Deploying Oracle Identity and Access Management cluster with Unicast configuration
  If multicast IP is disabled in deployment environment then you can deploy Oracle Identity and Access Management cluster with Unicast configuration.

Oracle Identity Governance Architecture

Oracle Identity Governance architecture consists of its components, runtime processes, process lifecycle, configuration artifacts, external dependencies, and log files.

Figure 6-1 shows the Oracle Identity Governance architecture:

Figure 6-1 Oracle Identity Governance Component Architecture



- Oracle Identity Governance Component Characteristics
- Runtime Processes
- Component and Process Lifecycle
- Starting and Stopping Oracle Identity Governance
- Configuration Artifacts
- External Dependencies
- Oracle Identity Governance Log File Locations

Oracle Identity Governance Component Characteristics

Oracle Identity Manager Server is Oracle's self-contained, standalone identity management solution. It provides User Administration, Workflow and Policy, Password Management, Audit and Compliance Management, User Provisioning and Organization and Role Management functionalities.

Oracle Identity Manager (OIM) is a standard Java EE application that is deployed on WebLogic Server and uses a database to store runtime and configuration data. The MDS schema contains configuration information; the runtime and user information is stored in the OIM schema.

OIM connects to the SOA Managed Servers over RMI to invoke SOA EJBs.

OIM uses the human workflow module of Oracle SOA Suite to manage its request workflow. OIM connects to SOA using the T3 URL for the SOA server, which is the front end URL for SOA. Oracle recommends using the load balancer or web server URL for clustered SOA servers. When the workflow completes, SOA calls back OIM web services using OIMFrontEndURL. Oracle SOA is deployed along with the OIM.

Several OIM modules use JMS queues. Each queue is processed by a separate Message Driven Bean (MDB), which is also part of the OIM application. Message producers are also part of the OIM application.

OIM uses a Quartz based scheduler for scheduled activities. Various scheduled activities occur in the background, such as disabling users after their end date.

In this release, BI Publisher is not embedded with OIM. However, you can integrate BI Publisher with OIM by following the instructions in Configuring Reports in *Developing and Customizing Applications for Oracle Identity Governance*.

Runtime Processes

Oracle Identity Manager deploys on WebLogic Server as a no-stage application. The OIM server initializes when the WebLogic Server it is deployed on starts up. As part of application initialization, the quartz-based scheduler is also started. Once initialization is done, the system is ready to receive requests from clients.

You must start the Design Console separately as a standalone utility.

Component and Process Lifecycle

Oracle Identity Manager deploys to a WebLogic Server as an externally managed application. By default, WebLogic Server starts, stops, monitors and manages other lifecycle events for the OIM application.

OIM starts after the application server components start. It uses the authenticator which is part of the OIM component mechanism; it starts up before the WebLogic JNDI initializes and the application starts.

OIM uses a Quartz technology-based scheduler that starts the scheduler thread on all WebLogic Server instances. It uses the database as centralized storage for picking and running scheduled activities. If one scheduler instance picks up a job, other instances do not pick up that same job.

You can configure Node Manager to monitor the server process and restart it in case of failure.

The Oracle Enterprise Manager Fusion Middleware Control is used to monitor as well as to modify the configuration of the application.

Starting and Stopping Oracle Identity Governance

You manage OIM lifecycle events with these command line tools and consoles:

- Oracle WebLogic Scripting Tool (WLST)
- WebLogic Server Administration Console
- Oracle Enterprise Manager Fusion Middleware Control
- Oracle WebLogic Node Manager

Configuration Artifacts

The OIM server configuration is stored in the MDS repository at `/db/oim-config.xml`. The `oim-config.xml` file is the main configuration file. Manage OIM configuration using the MBean browser through Oracle Enterprise Manager Fusion Middleware Control or command line MDS utilities. For more information about MDS utilities, see Migrating User Configurable Metadata Files in *Developing and Customizing Applications for Oracle Identity Governance*.

The installer configures JMS out-of-the-box; all necessary JMS queues, connection pools, data sources are configured on WebLogic application servers. These queues are created when OIM deploys:

- oimAttestationQueue
- oimAuditQueue
- oimDefaultQueue
- oimKernelQueue
- oimProcessQueue
- oimReconQueue
- oimSODQueue

The `xlconfig.xml` file stores Design Console and Remote Manager configuration.

External Dependencies

Oracle Identity Manager uses the Worklist and Human workflow modules of the Oracle SOA Suite for request flow management. OIM interacts with external repositories to store configuration and runtime data, and the repositories must be available during initialization and runtime. The OIM repository stores all OIM credentials. External components that OIM requires are:

- WebLogic Server
  - Administration Server
  - Managed Server
- Data Repositories

  - Configuration Repository (MDS Schema)
  - Runtime Repository (OIM Schema)
  - User Repository (OIM Schema)
  - SOA Repository (SOA Schema)
- BI Publisher, which can be optionally integrated with OIM

The Design Console is a tool used by the administrator for development and customization. The Design Console communicates directly with the OIM engine, so it relies on the same components that the OIM server relies on.

Remote Manager is an optional independent standalone application, which calls the custom APIs on the local system. It needs JAR files for custom APIs in its classpath.

Oracle Identity Governance Log File Locations

As a Java EE application deployed on WebLogic Server, all server log messages log to the server log file. OIM-specific messages log into the WebLogic Server diagnostic log file where the application is deployed.

WebLogic Server log files are in the directory:

```
DOMAIN_HOME/servers/serverName/logs
```

The three main log files are *serverName*.log, *serverName*.out, and *serverName*-diagnostic.log, where *serverName* is the name of the WebLogic Server. For example, if the WebLogic Server name is wls_OIM1, then the diagnostic log file name is wls_OIM1-diagnostic.log. Use Oracle Enterprise Manager Fusion Middleware Control to view log files.

Oracle Identity Governance High Availability Concepts

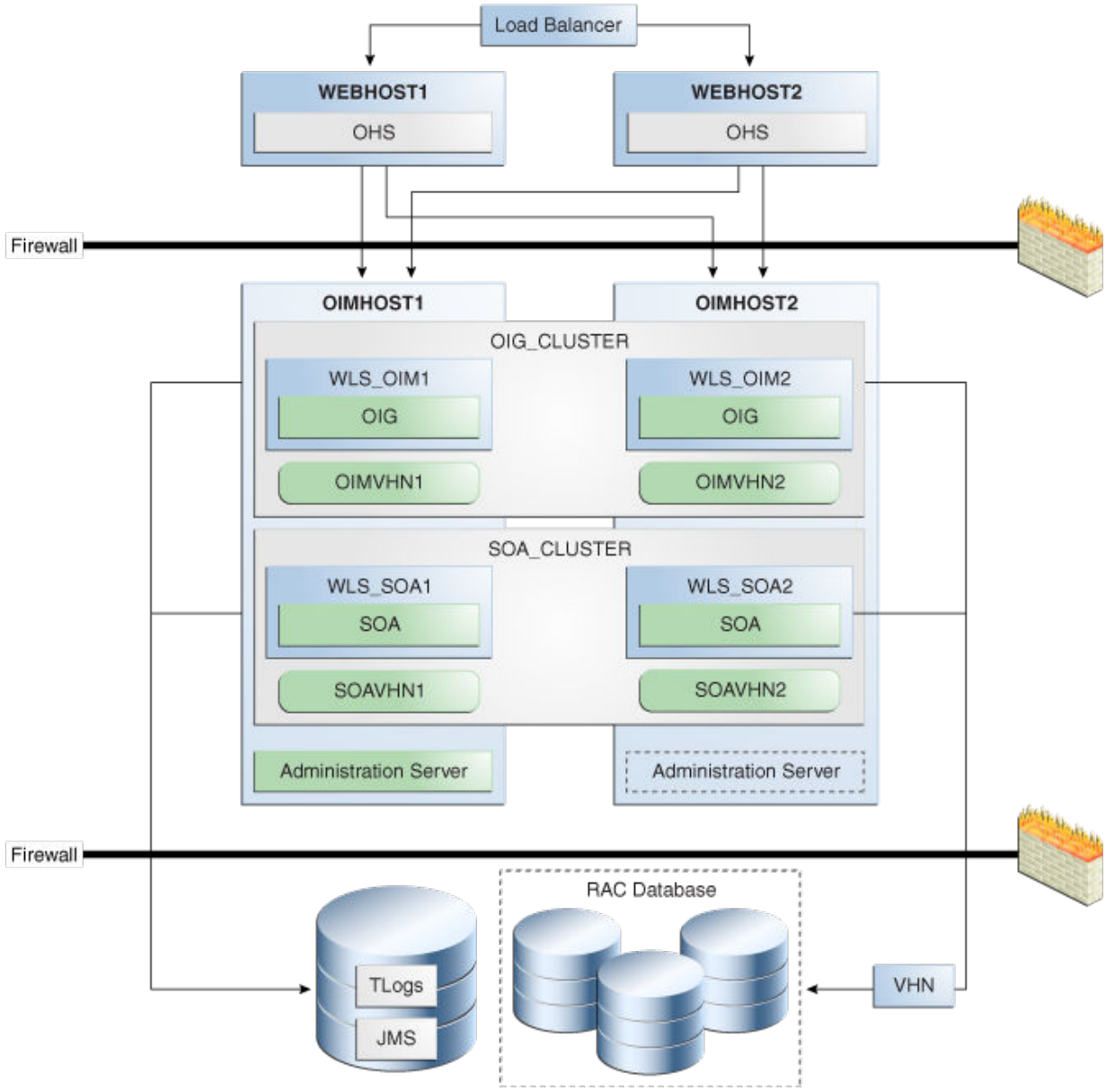
The concepts related to Oracle Identity Governance High Availability are OIG high availability architecture, starting and stopping OIG cluster, and cluster-wide configuration changes.

> **Note:**
>
> - You can deploy OIM on an Oracle RAC database, but Oracle RAC failover is not transparent for OIM in this release. If Oracle RAC failover occurs, end users may have to resubmit requests.
> - OIM always requires the availability of at least one node in the SOA cluster. If the SOA cluster is not available, end user requests fail. OIM does not retry for a failed SOA call. Therefore, the end user must retry when a SOA call fails.

Oracle Identity Governance High Availability Architecture

Figure 6-2 shows OIM deployed in a high availability architecture.

Figure 6-2 Oracle Identity Governance High Availability Architecture



On OIMHOST1, the following installations have been performed:

- An OIM instance is installed in the WLS_OIM1 Managed Server and a SOA instance is installed in the WLS_SOA1 Managed Server.
- The Oracle RAC database is configured in a GridLink data source to protect the instance from Oracle RAC node failure.
- A WebLogic Server Administration Server is installed. Under normal operations, this is the active Administration Server.

On OIMHOST2, the following installations have been performed:

- An OIM instance is installed in the WLS_OIM2 Managed Server and a SOA instance is installed in the WLS_SOA2 Managed Server.

- The Oracle RAC database is configured in a GridLink data source to protect the instance from Oracle RAC node failure.
- The instances in the WLS_OIM1 and WLS_OIM2 Managed Servers on OIMHOST1 and OIMHOST2 are configured as the OIM_Cluster cluster.
- The instances in the WLS_SOA1 and WLS_SOA2 Managed Servers on OIMHOST1 and OIMHOST2 are configured as the SOA_Cluster cluster.
- An Administration Server is installed. Under normal operations, this is the passive Administration Server. You make this Administration Server active if the Administration Server on OIMHOST1 becomes unavailable.

Figure 6-2 uses these virtual host names in the OIM high availability configuration:

- OIMVHN1 is the virtual hostname that maps to the listen address for the WLS_OIM1 Managed Server, and it fails over with server migration of the WLS_OIM1 Managed Server. It is enabled on the node where the WLS_OIM1 Managed Server is running (OIMHOST1 by default).
- OIMVHN2 is the virtual hostname that maps to the listen address for the WLS_OIM2 Managed Server, and it fails over with server migration of the WLS_OIM2 Managed Server. It is enabled on the node where the WLS_OIM2 Managed Server is running (OIMHOST2 by default).
- SOAVHN1 is the virtual hostname that is the listen address for the WLS_SOA1 Managed Server, and it fails over with server migration of the WLS_SOA1 Managed Server. It is enabled on the node where the WLS_SOA1 Managed Server is running (OIMHOST1 by default).
- SOAVHN2 is the virtual hostname that is the listen address for the WLS_SOA2 Managed Server, and it fails over with server migration of the WLS_SOA2 Managed Server. It is enabled on the node where the WLS_SOA2 Managed Server is running (OIMHOST2 by default).
- VHN refers to the virtual IP addresses for the Oracle Real Application Clusters (Oracle RAC) database hosts.

Starting and Stopping the OIG Cluster

By default, WebLogic Server starts, stops, monitors, and manages lifecycle events for the application. The OIM application leverages high availability features of clusters. In case of hardware or other failures, session state is available to other cluster nodes that can resume the work of the failed node.

Use these command line tools and consoles to manage OIM lifecycle events:

- WebLogic Server Administration Console
- Oracle Enterprise Manager Fusion Middleware Control
- Oracle WebLogic Scripting Tool (WLST)

Cluster-Wide Configuration Changes

For high availability environments, changing the configuration of one OIM instance changes the configuration of all the other instances, because all the OIM instances share the same configuration repository.

High Availability Directory Structure Prerequisites

A high availability deployment requires product installations and files to reside in specific directories. A standard directory structure makes facilitates configuration across nodes and product integration.

Before you configure high availability, verify that your environment meets the requirements that High Availability Directory Structure Prerequisites describes.

Oracle Identity Governance High Availability Configuration Steps

Oracle Identity Governance high availability configuration involves setting the prerequisites, configuring the domain, post-installation steps, starting servers, SOA integration, validating managed server instances, and scaling up and scaling out Oracle Identity Governance.

This section provides high-level instructions for setting up a high availability deployment for OIM and includes these topics:

- Prerequisites for Configuring Oracle Identity Governance
- Configuring the Domain
- Post-Installation Steps on OIMHOST1
- Starting the Administration Server, oim_server1, and soa_server1
- Integrating Oracle Identity Governance with Oracle SOA Suite
- Propagating Oracle Identity Governance to OIMHOST2
- Post-Installation Steps on OIMHOST2
- Validate Managed Server Instances on OIMHOST2
- Configuring Server Migration for OIG and SOA Managed Servers
- Configuring a Default Persistence Store for Transaction Recovery
- Install Oracle HTTP Server on WEBHOST1 and WEBHOST2
- Configuring Oracle Identity Governance to Work with the Web Tier
- Validate the Oracle HTTP Server Configuration
- Oracle Identity Governance Failover and Expected Behavior
- Scaling Up Oracle Identity Governance
- Scaling Out Oracle Identity Governance

Prerequisites for Configuring Oracle Identity Governance

Before you configure OIM for high availability, you must:

- Install the Oracle Database. See About Database Requirements for an Oracle Fusion Middleware Installation.

- Install the JDK on OIMHOST1 and OIMHOST2. See Preparing for Installation in *Installing and Configuring Oracle WebLogic Server and Coherence*.
- Install WebLogic Server, Oracle SOA Suite, and Oracle Identity Management software on OIMHOST1 and OIMHOST2 by using the quickstart installer. See Installing Oracle Identity Governance Using Quickstart Installer.
- Run the Repository Creation Utility to create the OIM schemas in a database. See Running RCU to Create the OIM Schemas in a Database.
- Running RCU to Create the OIM Schemas in a Database

Running RCU to Create the OIM Schemas in a Database

The schemas you create depend on the products you want to install and configure. Use a Repository Creation Utility (RCU) that is version compatible with the product you install. See Creating the Database Schemas.

Configuring the Domain

Use the Configuration Wizard to create and configure a domain.

See Configuring the Domain, Additional Domain Configuration, and Performing Post-Configuration Tasks for information about creating the Identity Management domain.

Post-Installation Steps on OIMHOST1

This section describes post-installation steps for OIMHOST1.

- Running the Offline Configuration Command
- Updating the System Properties for SSL Enabled Servers

Running the Offline Configuration Command

After you configure the Oracle Identity Governance domain, run the `offlineConfigManager` script to perform post configuration tasks.

Ensure that you run this command before you start any server. To run the `offlineConfigManager` command, do the following:

1. Set the following environment variables to the right values:
   - DOMAIN_HOME
   - JAVA_HOME
2. Ensure that you have execute permissions for the file *`OIM_HOME`*`/server/bin/offlineConfigManager.sh`.
3. Run the following command from the location *`OIM_HOME`*`/server/bin/`:
   - On Unix: `./offlineConfigManager.sh`
   - On Windows: `offlineConfigManager.bat`

Updating the System Properties for SSL Enabled Servers

For SSL enabled servers, you must set the required properties in the `setDomainEnv` file in the domain home.

Set the following properties in the *DOMAIN_HOME*`/bin/setDomainEnv.sh` (for UNIX) or *DOMAIN_HOME*`\bin\setDomainEnv.cmd` (for Windows) file before you start the servers:

- `-Dweblogic.security.SSL.ignoreHostnameVerification=true`
- `-Dweblogic.security.TrustKeyStore=DemoTrust`

Starting the Administration Server, oim_server1, and soa_server1

To start the Administration Server, oim_server1, and soa_server1:

1. To start the Administration Server, go to the DOMAIN_HOME/bin directory, and enter the following command:

   For UNIX: `./startWebLogic.sh`

   For Windows: `startWebLogic.cmd`

   If you selected Production Mode on the Domain Mode and JDK screen when you created the domain, you see a prompt for the Administrator user login credentials as the Administrator Account screen provides.
   You can verify that the Administration Server is up and running by accessing the Administration Server Console. The URL is provided on the End of Configuration screen (`http://administration_server_host:administration_server_port/console`). The default Administration Server port number is `7001`.

   > **Note:**
   >
   > Make sure that the database hosting your product schemas is up and running and accessible by the Administration Server.

2. Start the Node Manager on HOST1. To do so, run the following command from the *DOMAIN_HOME*/bin directory:

   (UNIX) Using `nohup` and `nm.out` as an example output file:

   `nohup ./startNodeManager.sh > LOG_DIR/nm.out&`

   Here, *LOG_DIR* is the location of directory in which you want to store the log files.

   (Windows) `startNodeManager.cmd`

3. Start the Oracle SOA Suite Managed Server(s) first and then the Oracle Identity Governance Managed Server(s). To start the Managed Servers:

   a. Login to Oracle Fusion Middleware Control:

      `http://administration_server_host:administration_server_port/em`

      The Enterprise Manager landing page lists servers configured for this domain and shows their status (such as Running or Shutdown). For a newly configured domain, only the **AdminServer(admin)** will be running.

b. Select wls_soa1.

c. From the Control list, select **Start**.

d. Repeat Steps b and c to start wls_oim1.

> **Note:**
>
> Ensure that you start the servers in the following order:
>
> i. Node Manager
>
> ii. Administration Server
>
> iii. Oracle SOA Suite Managed Server
>
> iv. Oracle Identity Manager Managed Server

e. On the main landing page, verify that all Managed Servers are up and running.

Integrating Oracle Identity Governance with Oracle SOA Suite

To integrate Oracle Identity Governance with Oracle SOA Suite:

1. Log in to Oracle Fusion Middleware Control by navigating to the following URL:

   http://administration_server_host:administration_server_port/em

2. Click **weblogic_domain**, and then select **System Mbean Browser**.

3. In the search box, enter `OIMSOAIntegrationMBean`, and click **Search**. The mbean is displayed.

> **Note:**
>
> If Oracle Identity Governance is still starting (coming up) or is just started (RUNNING MODE), then Enterprise Manager does not show any Mbeans defined by OIG. Wait for two minutes for the server to start, and then try searching for the Mbean in System Mbean Browser of the Enterprise Manager.

4. Click the **Operations** tab of mbean, and select **integrateWithSOAServer**.

5. Enter the required attributes, and then click **Invoke**.

Propagating Oracle Identity Governance to OIMHOST2

After the configuration succeeds on OIMHOST1, you can propagate it to OIMHOST2 by packing the domain on OIMHOST1 and unpacking it on OIMHOST2.

> **Note:**
>
> Oracle recommends that you perform a clean shut down of all Managed Servers on OIMHOST1 before you propagate the configuration to OIMHOST2.

To pack the domain on OIMHOST1 and unpack it on OIMHOST2:

1. On OIMHOST1, invoke the `pack` utility in the *ORACLE_HOME*`/oracle_common/common/bin` directory:

   ```
   pack.sh -domain=ORACLE_HOME/user_projects/domains/OIM_Domain -
   template =/u01/app/oracle/admin/templates/oim_domain.jar -
   template_name="OIM Domain" -managed=true
   ```

2. The previous step created the `oim_domain.jar` file in the following directory:

   ```
   /u01/app/oracle/admin/templates
   ```

   Copy `oim_domain.jar` from OIMHOST1 to a temporary directory on OIMHOST2.

3. On OIMHOST2, invoke the `unpack` utility in the *ORACLE_HOME*`/oracle_common/common/bin` directory and specify the `oim_domain.jar` file location in its temporary directory:

   ```
   unpack.sh -domain=ORACLE_HOME/user_projects/domains/OIM_Domain -
   template=/tmp/oim_domain.jar
   ```

Post-Installation Steps on OIMHOST2

Start Node Manager on OIMHOST2

Start the Node Manager on OIMHOST2 using the `startNodeManager.sh` script located under the following directory:

DOMAIN_HOME`/bin`

Start WLS_SOA2 and WLS_OIM2 Managed Servers on OIMHOST2

To start Managed Servers on OIMHOST2:

1. Start the WLS_SOA2 Managed Server using the Administration Console.
2. Start the WLS_OIM2 Managed Server using the Administration Console. The WLS_OIM2 Managed Server must be started after the WLS_SOA2 Managed Server is started.

Configuring SOA End Points

To configure the SOA end points:

1. Login to Oracle Fusion Middleware Enterprise Manager Control by navigating to the following URL:

   `http://`*ADMINISTRATION_SERVER_HOST*`:`*ADMINISTRATION_SERVER_PORT*`/em`

2. Click **weblogic_domain**, and then select **System Mbean Browser**.

3. Navigate to **Application Defined MBeans**, **oracle.iam**, **Server: *OIM_SERVER_NAME***, **Application: oim**, **XMLConfig:Config**, **XMLConfig.SOAConfig:SOAConfig**.
4. Enter values for the following attributes:

   SOA Config RMI URL: `cluster:t3://<SOA_cluster>`

   SOA Config SOAP URL: `http://<OHS_hostname>:<OHS_PORT>`

   > **Note:**
   >
   > If you are using a hierarchy of Oracle HTTP servers with a load balancer or web host on top to mitigate the single point of failure, then use the host and port of the exposed machine to access the OIM identity and sysadmin URLs.

5. Click **Invoke**.

Validate Managed Server Instances on OIMHOST2

Validate the Oracle Identity Manager (OIM) instances on OIMHOST2.

Open the OIM Console with this URL:

`http://identityvhn2.example.com:14000/identity`

Log in using the `xelsysadm` password.

Configuring Server Migration for OIG and SOA Managed Servers

For this high availability topology, Oracle recommends that you configure server migration for the WLS_OIM1, WLS_SOA1, WLS_OIM2, and WLS_SOA2 Managed Servers. See Section 3.9, "Whole Server Migration" for information on the benefits of using Whole Server Migration and why Oracle recommends it.

- The WLS_OIM1 and WLS_SOA1 Managed Servers on OIMHOST1 are configured to restart automatically on OIMHOST2 if a failure occurs on OIMHOST1.
- The WLS_OIM2 and WLS_SOA2 Managed Servers on OIMHOST2 are configured to restart automatically on OIMHOST1 if a failure occurs on OIMHOST2.

In this configuration, the WLS_OIM1, WLS_SOA1, WLS_OIM2 and WLS_SOA2 servers listen on specific floating IPs that WebLogic Server Migration fails over.

The subsequent topics enable server migration for the WLS_OIM1, WLS_SOA1, WLS_OIM2, and WLS_SOA2 Managed Servers, which in turn enables a Managed Server to fail over to another node if a server or process failure occurs.

- Editing Node Manager's Properties File
- Setting Environment and Superuser Privileges for the wlsifconfig.sh Script
- Configuring Server Migration Targets
- Testing the Server Migration

Editing Node Manager's Properties File

You must edit the `nodemanager.properties` file to add the following properties for each node where you configure server migration:

```
Interface=eth0
eth0=*,NetMask=255.255.248.0
UseMACBroadcast=true
```

- `Interface`: Specifies the interface name for the floating IP (such as `eth0`).

  > **Note:**
  >
  > Do not specify the sub interface, such as `eth0:1` or `eth0:2`. This interface is to be used without the `:0`, or `:1`. The Node Manager's scripts traverse the different `:X` enabled IPs to determine which to add or remove. For example, valid values in Linux environments are `eth0`, `eth1`, or, `eth2`, `eth3`, `eth`*n*, depending on the number of interfaces configured.

- `NetMask`: Net mask for the interface for the floating IP. The net mask should be the same as the net mask on the interface; `255.255.255.0` is an example. The actual value depends on your network.
- `UseMACBroadcast`: Specifies whether or not to use a node's MAC address when sending ARP packets, that is, whether or not to use the `-b` flag in the `arping` command.

Verify in Node Manager's output (shell where Node Manager starts) that these properties are being used or problems may arise during migration. (Node Manager must be restarted to do this.) You should see an entry similar to the following in Node Manager's output:

```
...
StateCheckInterval=500
Interface=eth0
NetMask=255.255.255.0
...
```

Setting Environment and Superuser Privileges for the wlsifconfig.sh Script

To set environment and superuser privileges for the `wlsifconfig.sh` script for each node where you configure server migration:

1. Modify the login profile of the user account that you use to run Node Manager to ensure that the PATH environment variable for the Node Manager process includes directories housing the `wlsifconfig.sh` and `wlscontrol.sh` scripts, and the `nodemanager.domains` configuration file. Ensure that your PATH environment variable includes these files:

Table 6-1 Files Required for the PATH Environment Variable

File	Located in this directory
wlsifconfig.sh	*DOMAIN_HOME*/bin/server_migration
wlscontrol.sh	*WL_HOME*/common/bin
nodemanager.domains	*WL_HOME*/common

2. Grant sudo configuration for the `wlsifconfig.sh` script.
   - Configure sudo to work without a password prompt.
   - For security reasons, Oracle recommends restricting to the subset of commands required to run the `wlsifconfig.sh` script. For example, perform the following steps to set the environment and superuser privileges for the `wlsifconfig.sh` script:
   - Grant sudo privilege to the WebLogic user (`oracle`) with no password restriction, and grant execute privilege on the `/sbin/ifconfig` and `/sbin/arping binaries`.
   - Ensure that the script is executable by the WebLogic user. The following is an example of an entry inside `/etc/sudoers` granting sudo execution privilege for `oracle` and also over `ifconfig` and `arping`:

     ```
     oracle ALL=NOPASSWD: /sbin/ifconfig,/sbin/arping
     ```

> **Note:**
>
> Ask the system administrator for the sudo and system rights as appropriate to this step.

Configuring Server Migration Targets

You first assign all available nodes for the cluster's members and then specify candidate machines (in order of preference) for each server that is configured with server migration. To configure cluster migration in a cluster:

1. Log into the Administration Console.
2. In the Domain Structure window, expand **Environment** and select **Clusters**.
3. Click the cluster you want to configure migration for in the Name column.
4. Click the **Migration** tab.
5. Click **Lock and Edit**.
6. In the **Available** field, select the machine to which to enable migration and click the right arrow.
7. Select the data source to use for automatic migration. In this case, select the leasing data source, which is `WLSSchemaDataSource`.
8. Click **Save**.
9. Click **Activate Changes**.
10. Set the candidate machines for server migration. You must perform this task for all Managed Servers as follows:

a. In the Domain Structure window of the Administration Console, expand **Environment** and select **Servers**.

> **Tip:**
>
> Click **Customize this table** in the Summary of Servers page and move Current Machine from the Available window to the Chosen window to view the machine that the server runs on. This will be different from the configuration if the server migrates automatically.

b. Select the server that you want to configure migration for.

c. Click the **Migration** tab.

d. In the **Available** field, located in the Migration Configuration section, select the machines you want to enable migration to and click the right arrow.

e. Select **Automatic Server Migration Enabled**. This enables Node Manager to start a failed server on the target node automatically.

f. Click **Save** then Click **Activate Changes**.

g. Repeat the steps above for any additional Managed Servers.

h. Restart the administration server, Node Managers, and the servers for which server migration has been configured.

Testing the Server Migration

To verify that server migration works properly:

From OIMHOST1:

1. Stop the WLS_OIM1 Managed Server by running the command:

   ```
   OIMHOST1> kill -9 pid
   ```

   where *pid* specifies the process ID of the Managed Server. You can identify the pid in the node by running this command:

   ```
   OIMHOST1> ps -ef | grep WLS_OIM1
   ```

2. Watch the Node Manager console. You should see a message indicating that WLS_OIM1's floating IP has been disabled.
3. Wait for Node Manager to try a second restart of WLS_OIM1. It waits for a fence period of 30 seconds before trying this restart.
4. Once Node Manager restarts the server, stop it again. Node Manager should now log a message indicating that the server will not be restarted again locally.

From OIMHOST2:

1. Watch the local Node Manager console. After 30 seconds since the last try to restart WLS_OIM1 on OIMHOST1, Node Manager on OIMHOST2 should prompt that the floating IP for WLS_OIM1 is being brought up and that the server is being restarted in this node.
2. Access the soa-infra console in the same IP.

Follow the steps above to test server migration for the WLS_OIM2, WLS_SOA1, and WLS_SOA2 Managed Servers.

Table 6-2 shows the Managed Servers and the hosts they migrate to in case of a failure.

Table 6-2 WLS_OIM1, WLS_OIM2, WLS_SOA1, WLS_SOA2 Server Migration

Managed Server	Migrated From	Migrated To
WLS_OIM1	OIMHOST1	OIMHOST2
WLS_OIM2	OIMHOST2	OIMHOST1
WLS_SOA1	OIMHOST1	OIMHOST2
WLS_SOA2	OIMHOST2	OIMHOST1

From Verification From the Administration Console

To verify migration in the Administration Console:

1. Log into the Administration Console at http://oimhost1.example.com:7001/console using administrator credentials.
2. Click **Domain** on the left console.
3. Click the **Monitoring** tab and then the **Migration** sub tab.

   The Migration Status table provides information on the status of the migration.

> **Note:**
>
> After a server migrates, to fail it back to its original node/machine, stop the Managed Server in the Administration Console then start it again. The appropriate Node Manager starts the Managed Server on the machine it was originally assigned to.

Configuring a Default Persistence Store for Transaction Recovery

Each Managed Server has a transaction log that stores information about in-flight transactions that the Managed Server coordinates that may not complete. WebLogic Server uses the transaction log to recover from system/network failures. To leverage the Transaction Recovery Service migration capability, store the transaction log in a location that all Managed Servers in a cluster can access. Without shared storage, other servers in the cluster can't run transaction recovery in the event of a server failure, so the operation may need to be retried.

> **Note:**
>
> Oracle recommends a location on a Network Attached Storage (NAS) device or Storage Area Network (SAN).

To set the location for default persistence stores for the OIM and SOA Servers:

1. Log into the Administration Console at http://oimhost1.*example*.com:7001/console using administrator credentials.
2. In the Domain Structure window, expand the **Environment** node and then click the **Servers** node. The Summary of Servers page opens.
3. Select the name of the server (represented as a hyperlink) in the **Name** column of the table. The Settings page for the server opens to the Configuration tab.
4. Select the **Services** subtab of the Configuration tab (not the Services top-level tab).
5. In the Default Store section, enter the path to the folder where the default persistent stores store their data files. The directory structure of the path should be:
   - For the WLS_SOA1 and WLS_SOA2 servers, use a directory structure similar to:

     ORACLE_BASE/admin/*domainName*/*soaClusterName*/tlogs
   - For the WLS_OIM1 and WLS_OIM2 servers, use a directory structure similar to:

     ORACLE_BASE/admin/*domainName*/*oimClusterName*/tlogs
6. Click **Save**.

> **Note:**
>
> To enable migration of Transaction Recovery Service, specify a location on a persistent storage solution that is available to the Managed Servers in the cluster. WLS_SOA1, WLS_SOA2, WLS_OIM1, and WLS_OIM2 must be able to access this directory.

Install Oracle HTTP Server on WEBHOST1 and WEBHOST2

Install Oracle HTTP Server on WEBHOST1 and WEBHOST2.

Configuring Oracle Identity Governance to Work with the Web Tier

You can co-locate Oracle HTTP Server and Oracle Identity Governance in a High Availability set up in the following ways:

- Create an OIG domain and extend same OIM domain with Oracle HTTP Server. In this case, you can re-use the same schema for both Oracle HTTP Server and OIG.
- Create a separate domain for Oracle HTTP Server and OIG. In this case, you can not re-use the same schema for both Oracle HTTP Server and OIG. You will need two separate schemas.
- Create separate domains for Oracle HTTP Server and OIG using Silent/wlst. In this case, you can re-use the same schema for both Oracle HTTP Server and OIG.

The following topics describe how to configure OIG to work with the Oracle Web Tier.

- Prerequisites to Configure OIG to Work with the Web Tier
- Configuring SSL Certificates for Load Balancer
  For SSL enabled server with load balancer, you must configure SSL certificates when using Oracle Web Tier for `OIMExternalFrontendURL` .
- Configuring Oracle HTTP Servers to Front End OIM, and SOA Managed Servers

Prerequisites to Configure OIG to Work with the Web Tier

Verify that the following tasks have been performed:

1. Oracle Web Tier has been installed on WEBHOST1 and WEBHOST2.
2. OIM is installed and configured on OIMHOST1 and OIMHOST2.
3. The load balancer has been configured with a virtual hostname (`sso.example.com`) pointing to the web servers on WEBHOST1 and WEBHOST2. I`sso.example.com`is customer facing and the main point of entry; it is typically SSL terminated.
4. The load balancer has been configured with a virtual hostname (`oiminternal.example.com`) pointing to web servers WEBHOST1 and WEBHOST2. `oiminternal.example.com` is for internal callbacks and is *not* customer facing.

Configuring SSL Certificates for Load Balancer

For SSL enabled server with load balancer, you must configure SSL certificates when using Oracle Web Tier for `OIMExternalFrontendURL` .

1. Export the SSL certificates from load balancer.
2. Import the SSL certificates into the DemoTrust / `cacerts` keystores. If your not using default stores, import the SSL certificates into your custom stores.
3. Shutdown both the SOA and OIM servers.
4. Clear the server cache and temporary directories.
5. Restart both the SOA and OIM servers.

> **Note:**
>
> For more information about Keystores, see About Configuring Keystores in WebLogic Server

Configuring Oracle HTTP Servers to Front End OIM, and SOA Managed Servers

1. On each of the web servers on `WEBHOST1` and `WEBHOST2`, create a file named `mod_wls_ohs.conf` in the directory *`OHS_DOMAIN_HOME`*`/config/fmwconfig/components/OHS/instances/`*`OHS_INSTANCE_NAME`*.

   This file must contain the following information:

```
# oim admin console(idmshell based)
   <Location /admin>
    SetHandler weblogic-handler
    WLCookieName    oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
```

```
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
   </Location>

# oim self and advanced admin webapp consoles(canonic webapp)

  <Location /oim>
    SetHandler weblogic-handler
    WLCookieName    oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
   </Location>

  <Location /identity>
    SetHandler weblogic-handler
    WLCookieName    oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
    </Location>

  <Location /sysadmin>
    SetHandler weblogic-handler
    WLCookieName    oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
    </Location>

# SOA Callback webservice for SOD - Provide the SOA Managed Server Ports
  <Location /sodcheck>
    SetHandler weblogic-handler
    WLCookieName    oimjsessionid
    WebLogicCluster soavhn1.example.com:7003,soavhn2.example.com:7003
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
   </Location>

# Callback webservice for SOA. SOA calls this when a request is approved/
rejected
# Provide the OIM Managed Server Port
  <Location /workflowservice>
    SetHandler weblogic-handler
    WLCookieName    oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>
# used for FA Callback service.
  <Location /callbackResponseService>
    SetHandler weblogic-handler
    WLCookieName    oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
```

```
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>

# spml xsd profile
  <Location /spml-xsd>
    SetHandler weblogic-handler
    WLCookieName    oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>

  <Location /HTTPClnt>
    SetHandler weblogic-handler
    WLCookieName    oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>


  <Location /reqsvc>
    SetHandler weblogic-handler
    WLCookieName oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>


  <Location /integration>
    SetHandler weblogic-handler
    WLCookieName oimjsessionid
    WebLogicCluster soavhn1.example.com:7003,soavhn2.example.com:7003
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>


  <Location /provisioning-callback>
    SetHandler weblogic-handler
    WLCookieName oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>

  <Location /CertificationCallbackService>
   SetHandler weblogic-handler
   WLCookieName JSESSIONID
   WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
   WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   WLProxySSL ON
   WLProxySSLPassThrough ON
 </Location>
```

```
<Location /ucs>
 SetHandler weblogic-handler
 WLCookieName oimjsessionid
 WebLogicCluster soavhn1.example.com:7003,soavhn2.example.com:7003
 WLLogFile /tmp/web_log.log
 WLProxySSL ON
 WLProxySSLPassThrough ON
</Location>

<Location /FacadeWebApp>
 SetHandler weblogic-handler
 WLCookieName oimjsessionid
 WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
 WLLogFile /tmp/web_log.log
 WLProxySSL ON
 WLProxySSLPassThrough ON
</Location>

<Location /iam/governance/configmgmt>
 SetHandler weblogic-handler
 WLCookieName oimjsessionid
 WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
 WLLogFile /tmp/web_log.log
 WLProxySSL ON
 WLProxySSLPassThrough ON
</Location>

<Location /iam/governance/scim/v1>
 SetHandler weblogic-handler
 WLCookieName oimjsessionid
 WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
 WLLogFile /tmp/web_log.log
 WLProxySSL ON
 WLProxySSLPassThrough ON>
</Location>

<Location /iam/governance/token/api/v1>
 SetHandler weblogic-handler
 WLCookieName oimjsessionid
 WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
 WLLogFile /tmp/web_log.log
 WLProxySSL ON
 WLProxySSLPassThrough ON
</Location>

<Location /OIGUI>
 SetHandler weblogic-handler
 WLCookieName oimjsessionid
 WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
 WLLogFile /tmp/web_log.log
 WLProxySSL ON
 WLProxySSLPassThrough ON
</Location>

<Location /iam/governance/applicationmanagement>
 SetHandler weblogic-handler
 WLCookieName oimjsessionid
 WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
 WLLogFile /tmp/web_log.log
 WLProxySSL ON
 WLProxySSLPassThrough ON
```

```
</Location>

<Location /iam/governance/adminservice/api/v1>
 SetHandler weblogic-handler
 WLCookieName oimjsessionid
 WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
 WLLogFile /tmp/web_log.log
 WLProxySSL ON
 WLProxySSLPassThrough ON
</Location>

<Location /iam/governance/selfservice/api/v1>
 SetHandler weblogic-handler
 WLCookieName oimjsessionid
 WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
 WLLogFile /tmp/web_log.log
 WLProxySSL ON
 WLProxySSLPassThrough ON
</Location>
```

2. Create a file called `virtual_hosts.conf` in *ORACLE_INSTANCE*`/config/OHS/`*COMPONENT*`/moduleconf`. The file must contain the following information:

> **Note:**
>
> *COMPONENT* is typically `ohs1` or `ohs2`. However, the name depends on choices you made during OHS installation.

```
NameVirtualHost *:7777
<VirtualHost *:7777>

  ServerName http://sso.example.com:7777
  RewriteEngine On
  RewriteOptions inherit
  UseCanonicalName On
  </VirtualHost>

<VirtualHost *:7777>
  ServerName http://oiminternal.example.com:80
  RewriteEngine On
  RewriteOptions inherit
  UseCanonicalName On
</VirtualHost>
```

3. Save the file on both `WEBHOST1` and `WEBHOST2`.
4. Stop and start the Oracle HTTP Server instances on both `WEBHOST1` and `WEBHOST2`.

Validate the Oracle HTTP Server Configuration

To validate that Oracle HTTP Server is configured properly, follow these steps:

1. In a web browser, enter the following URL for the Oracle Identity Manager Console:

   ```
   http://sso.example.com:7777/identity
   ```

   The Oracle Identity Manager Console login page should display.

2. Log into the Oracle Identity Manager Console using the credentials for the `xelsysadm` user.

Oracle Identity Governance Failover and Expected Behavior

In a high availability environment, you configure Node Manager to monitor Oracle WebLogic Servers. In case of failure, Node Manager restarts the WebLogic Server.

A hardware load balancer load balances requests between multiple OIM instances. If one OIM Managed Server fails, the load balancer detects the failure and routes requests to surviving instances.

In a high availability environment, state and configuration information is stored in a database that all cluster members share. Surviving OIM instances continue to seamlessly process any unfinished transactions started on the failed instance because state information is in the shared database, available to all cluster members.

When an OIM instance fails, its database and LDAP connections are released. Surviving instances in the active-active deployment make their own connections to continue processing unfinished transactions on the failed instance.

When you deploy OIM in a high availability configuration:

- You can deploy OIM on an Oracle RAC database, but Oracle RAC failover is not transparent for OIM in this release. If Oracle RAC failover occurs, end users may have to resubmit their requests.
- Oracle Identity Manager always requires the availability of at least one node in the SOA cluster. If the SOA cluster is not available, end user requests fail. OIM does not retry for a failed SOA call. Therefore, the end user must retry when a SOA call fails.

Scaling Up Oracle Identity Governance

You can scale out or scale up the OIG high availability topology. When you *scale up* the topology, you add new Managed Servers to nodes that are already running one or more Managed Servers. When you *scale out* the topology, you add new Managed Servers to new nodes. See Scaling Out Oracle Identity Governance to scale out.

In this case, you have a node that runs a Managed Server configured with SOA. The node contains:

- A Middleware home
- An Oracle HOME (SOA)
- A domain directory for existing Managed Servers

You can use the existing installations (Middleware home and domain directories) to create new WLS_OIM and WLS_SOA Managed Servers. You do not need to install OIM and SOA binaries in a new location or run pack and unpack.

This procedure describes how to clone OIM and SOA Managed Servers. You may clone one or two of these component types, as long as one of them is OIM.

Note the following:

- This procedure refers to WLS_OIM and WLS_SOA. However, you may not be scaling up both the components. For each step, choose the component(s) that you

are scaling up in your environment. Also, some steps do not apply to all components

- The persistent store's shared storage directory for JMS Servers must exist before you start the Managed Server or the start operation fails.
- Each time you specify the persistent store's path, it must be a directory on shared storage

To scale up the topology:

1. In the Administration Console, clone WLS_OIM1/WLS_SOA1. The Managed Server that you clone should be one that already exists on the node where you want to run the new Managed Server.
   a. Select **Environment -> Servers** from the Administration Console.
   b. Select the Managed Server(s) that you want to clone.
   c. Select **Clone**.
   d. Name the new Managed Server WLS_OIM`n`/WLS_SOA`n`, where `n` is a number to identity the new Managed Server.

   The rest of the steps assume that you are adding a new Managed Server to OIMHOST1, which is already running WLS_OIM1 and WLS_SOA1.
2. For the listen address, assign the hostname or IP for the new Managed Server(s). If you plan to use server migration, use the VIP (floating IP) to enable Managed Server(s) to move to another node. Use a VIP different from the VIP that the existing Managed Server uses.
3. Create JMS Servers for OIM/SOA, BPM, UMS, JRFWSAsync, and SOAJMServer on the new Managed Server.
   a. In the Administration Console, create a new persistent store for the OIM JMS Server(s) and name it. Specify the store's path, a directory on shared storage.

      ORACLE_BASE/admin/*domain_name*/*cluster_name*/jms
   b. Create a new JMS Server for OIM. Use `JMSFileStore_`*n* for JMSServer. Target `JMSServer_`*n* to the new Managed Server(s).
   c. Create a persistence store for the new UMSJMSServer(s), for example, `UMSJMSFileStore_`*n*. Specify the store's path, a directory on shared storage.

      ORACLE_BASE/admin/*domain_name*/*cluster_name*/jms/UMSJMSFileStore_n
   d. Create a new JMS Server for UMS, for example, `UMSJMSServer_`*n*. Target it to the new Managed Server (WLS_SOAn).
   e. Create a persistence store for the new BPMJMSServer(s), for example, `BPMJMSFileStore_`*n*. Specify the store's path, a directory on shared storage.

      ORACLE_BASE/admin/*domain_name*/*cluster_name*/jms/BPMJMSFileStore_n
   f. Create a new JMS Server for BPM, for example, `BPMJMSServer_`*n*. Target it to the new Managed Server (WLS_SOAn).
   g. Create a new persistence store for the new JRFWSAsyncJMSServer, for example, `JRFWSAsyncJMSFileStore_`*n*. Specify the store's path, a directory on shared storage.

      ORACLE_BASE/admin/*domain_name*/*cluster_name*/jms/JRFWSAsyncJMSFileStore_n
   h. Create a JMS Server for JRFWSAsync, for example, `JRFWSAsyncJMSServer_`*n*. Use `JRFWSAsyncJMSFileStore_`*n* for this JMSServer. Target `JRFWSAsyncJMSServer_`*n* to the new Managed Server (WLS_OIMn).

> **Note:**
>
> You can also assign `SOAJMSFileStore_n` as store for the new JRFWSAsync JMS Servers. For clarity and isolation, individual persistent stores are used in the following steps.

i. Create a persistence store for the new SOAJMSServer, for example, `SOAJMSFileStore_auto_n`. Specify the store's path, a directory on shared storage.

   ORACLE_BASE`/admin/`*domain_name*`/`*cluster_name*`/jms/SOAJMSFileStore_auto_`*n*

j. Create a JMS Server for SOA, for example, `SOAJMSServer_auto_n`. Use `SOAJMSFileStore_auto_n` for this JMSServer. Target `SOAJMSServer_auto_n` to the new Managed Server (WLS_SOAn).

> **Note:**
>
> You can also assign `SOAJMSFileStore_n` as store for the new PS6 JMS Servers. For the purpose of clarity and isolation, individual persistent stores are used in the following steps.

k. Update SubDeployment targets for SOA JMS Module to include the new SOA JMS Server. Expand the **Services** node, then expand the **Messaging** node. Choose **JMS Modules** from the Domain Structure window. Click **SOAJMSModule** (hyperlink in the **Names** column). In the Settings page, click the **SubDeployments** tab. In the subdeployment module, click the **SOAJMSServerXXXXXX** subdeployment and add `SOAJMSServer_n` to it Click **Save**.

> **Note:**
>
> A subdeployment module name is a random name in the form *COMPONENT*JMSServerXXXXXX. It comes from the Configuration Wizard JMS configuration for the first two Managed Servers, WLS_*COMPONENT*1 and WLS_*COMPONENT*2).

l. Update SubDeployment targets for UMSJMSSystemResource to include the new UMS JMS Server. Expand the **Services** node, then expand the **Messaging** node. Choose **JMS Modules** from the Domain Structure window. Click **UMSJMSSystemResource** (hyperlink in the Names column). In the Settings page, click the SubDeployments tab. In the subdeployment module, click the **UMSJMSServerXXXXXX** subdeployment and add `UMSJMSServer_n` to it. Click **Save**.

m. Update SubDeployment targets for OIMJMSModule to include the new OIM JMS Server. Expand the **Services** node, then expand **Messaging** node. Choose **JMS Modules** from the Domain Structure window. Click **OIMJMSModule** (hyperlink in the **Names** column). In the Settings page, click the SubDeployments tab. In the subdeployment module, click **OIMJMSServerXXXXXX** and `OIMJMSServer_n` to it. Click **Save**.

n. Update SubDeployment targets for the JRFWSAsyncJmsModule to include the new JRFWSAsync JMS Server. Expand the **Services** node then expand the **Messaging** node. Choose **JMS Modules** from the Domain Structure window. Click **JRFWSAsyncJmsModule** (hyperlink in the Names column of the table). In the Settings page, click the SubDeployments tab. Click the **JRFWSAsyncJMSServerXXXXXX** subdeployment and add `JRFWSAsyncJMSServer_`*n* to this subdeployment. Click **Save**

o. Update SubDeployment targets for BPM JMS Module to include the new BPM JMS Server. Expand the **Services** node, then expand the **Messaging** node. Choose **JMS Modules** from the Domain Structure window. Click **BPMJMSModule** (hyperlink in the **Names** column). In the Settings page, click the **SubDeployments** tab. In the subdeployment module, click the **BPMJMSServerXXXXXX** subdeployment and add `BPMJMSServer_`*n* to it. Click **Save**.

4. Configure the transaction persistent store for the new server in a shared storage location visible from other nodes.

   From the Administration Console, select **Server_name > Services** tab. Under Default Store, in Directory, enter the path to the default persistent store.

5. Disable hostname verification for the new Managed Server (required before starting/ verifying a WLS_SOA*n* Managed Server) You can re-enable it after you configure server certificates for Administration Server / Node Manager communication in SOAHOSTn. If the source server (from which you cloned the new Managed Server) had disabled hostname verification, these steps are not required; hostname verification settings propagate to a cloned server.

   To disable hostname verification:

   a. In the Administration Console, expand the **Environment** node in the Domain Structure window.

   b. Click **Servers**. Select WLS_SOAn in the **Names** column of the table.

   c. Click the SSL tab. Click **Advanced.**

   d. Set **Hostname Verification** to **None**. Click **Save**.

6. Start and test the new Managed Server from the Administration Console.

   a. Shut down the existing Managed Servers in the cluster.

   b. Ensure that the newly created Managed Server is up.

   c. Access the application on the newly created Managed Server to verify that it works. A login page opens for OIM. For SOA, a HTTP basic authorization opens.

Table 6-3 Managed Server Test URLs

Component	Managed Server Test URL
SOA	http://*vip*:*port*/soa-infra
OIM	http://*vip*:*port*/identity

7. In the Administration Console, select **Services** then **Foreign JNDI provider**. Confirm that **ForeignJNDIProvider-SOA** targets `cluster:t3://soa_cluster`, not a Managed Server(s). You target the cluster so that new Managed Servers don't require configuration. If **ForeignJNDIProvider-SOA** does not target the cluster, target it to the cluster.

8. Configure Server Migration for the new Managed Server.

> **Note:**
>
> For scale up, the node must have a Node Manager, an environment configured for server migration, and the floating IP for the new Managed Server(s).

To configure server migration:

a. Log into the Administration Console.

b. In the left pane, expand **Environment** and select **Servers**.

c. Select the server (hyperlink) that you want to configure migration for.

d. Click the Migration tab.

e. In the Available field, in the Migration Configuration section, select machines to enable migration for and click the right arrow. Select the same migration targets as for the servers that already exist on the node.

   For example:

   For new Managed Servers on SOAHOST1, which is already running WLS_SOA1, select SOAHOST2.

   For new Managed Servers on SOAHOST2, which is already running WLS_SOA2, select SOAHOST1.

   Verify that the appropriate resources are available to run Managed Servers concurrently during migration.

f. Select the **Automatic Server Migration Enabled** option to enable Node Manager to start a failed server on the target node automatically.

g. Click **Save**.

h. Restart the Administration Server, Managed Servers, and Node Manager.

i. Repeat these steps to configure server migration for the newly created WLS_OIMn Managed Server.

9. To test server migration for this new server, follow these steps from the node where you added the new server:

   a. Stop the Managed Server.

      Run `kill -9` *pid* on the PID of the Managed Server. To identify the PID of the node, enter, for example, `ps -ef | grep WLS_SOA`*n*.

   b. Watch Node Manager Console for a message indicating that the Managed Server floating IP is disabled.

   c. Wait for Node Manager to try a second restart of the Managed Server. Node Manager waits for 30 seconds before trying this restart.

   d. After Node Manager restarts the server, stop it again. Node Manager logs a message indicating that the server will not restart again locally.

10. Edit the OHS configuration file to add the new managed server(s). See Configuring Oracle HTTP Server to Recognize New Managed Servers.

Scaling Out Oracle Identity Governance

When you scale out the topology, you add new Managed Servers configured with software to new nodes.

> **Note:**
>
> Steps in this procedure refer to WLS_OIM and WLS_SOA. However, you may not be scaling up both the components. For each step, choose the component(s) that you are scaling up in your environment. Some steps do not apply to all components.

Before you scale out, check that you meet these requirements:

- Existing nodes running Managed Servers configured with OIM and SOA in the topology.
- The new node can access existing home directories for WebLogic Server, SOA, and OIM. (Use existing installations in shared storage to create new Managed Server. You do not need to install WebLogic Server or component binaries in a new location, but must run pack and unpack to bootstrap the domain configuration in the new node.)

  > **Note:**
  >
  > If there is no existing installation in shared storage, you must install WebLogic Server, SOA, and OIM in the new nodes.

  > **Note:**
  >
  > When multiple servers in different nodes share *ORACLE_HOME* or *WL_HOME*, Oracle recommends keeping the Oracle Inventory and Middleware home list in those nodes updated for consistency in the installations and application of patches. To update the oraInventory in a node and "attach" an installation in a shared storage to it, use *ORACLE_HOME*`/oui/bin/attachHome.sh`.

To scale out the topology:

1. On the new node, mount the existing Middleware home, which contains the SOA and OIG installations, and ensure that the new node has access to this directory, just like the rest of the nodes in the domain.
2. Attach *ORACLE_HOME* in shared storage to the local Oracle Inventory. For example:

   ```
   cd /u01/app/oracle/soa/
   ./attachHome.sh -jreLoc u01/app/JRE-JDK_version>
   ```

   To update the Middleware home list, create (or edit, if another WebLogic installation exists in the node) the *ORACLE_HOME*`/bea/beahomelist` file and add `u01/app/oracle` to it.
3. Log in to the Administration Console.

4. Create a new machine for the new node. Add the machine to the domain.
5. Update the machine's Node Manager's address to map the IP of the node that is being used for scale out.
6. Clone WLS_OIM1/WLS_SOA1.

   To clone OIM and SOA:

   a. Select **Environment -> Servers** from the Administration Console.
   b. Select the Managed Server(s) that you want to clone.
   c. Select **Clone**.
   d. Name the new Managed Server WLS_OIM`n`/WLS_SOA`n`, where `n` is a number to identity the new Managed Server.

> **Note:**
>
> These steps assume that you are adding a new server to node *n*, where no Managed Server was running previously.

7. Assign the hostname or IP to use for the new Managed Server for the listen address of the Managed Server. In addition, update the value of the Machine parameter with the new machine created in step 4.

   If you plan to use server migration for this server (which Oracle recommends), this should be the server VIP (*floating IP*). This VIP should be different from the one used for the existing Managed Server.
8. Create JMS servers for OIM (if applicable), UMS, BPM, JRFWSAsync, and SOA on the new Managed Server.

   a. In the Administration Console, create a new persistent store for the OIM JMS Server and rename it. Specify the store's path, a directory on shared storage.

      ORACLE_BASE/admin/*domain_name*/*cluster_name*/jms
   b. Create a new JMS Server for OIM. Use `JMSFileStore_`*n* for JMSServer. Target `JMSServer_`*n* to the new Managed Server(s).
   c. Create a persistence store for the new UMSJMSServer(s), for example, `UMSJMSFileStore_`*n*. Specify the store's path, a directory on shared storage.

      ORACLE_BASE/admin/*domain_name*/*cluster_name*/jms/UMSJMSFileStore_n
   d. Create a new JMS Server for UMS, for example, `UMSJMSServer_`*n*. Target it to the new Managed Server, which is WLS_SOAn (migratable).
   e. Create a persistence store for the new BPMJMSServer(s), for example, `BPMJMSFileStore_`*n*. Specify the store's path, a directory on shared storage.

      ORACLE_BASE/admin/*domain_name*/*cluster_name*/jms/BPMJMSFileStore_*n*
   f. Create a new JMS Server for BPM, for example, `BPMJMSServer_`*n*. Target it to the new Managed Server, which is WLS_SOAn (migratable).
   g. Create a new persistence store for the new JRFWSAsyncJMSServer, for example, `JRFWSAsyncJMSFileStore_`*n*. Specify the store's path, a directory on shared storage.

      ORACLE_BASE/admin/*domain_name*/*cluster_name*/jms/JRFWSAsyncJMSFileStore_*n*

h. Create a JMS Server for JRFWSAsync, for example, `JRFWSAsyncJMSServer_n`. Use `JRFWSAsyncJMSFileStore_n` for this JMSServer. Target `JRFWSAsyncJMSServer_n` to the new Managed Server, which is WLS_OIMn (migratable).

> **Note:**
>
> You can also assign `SOAJMSFileStore_n` as store for the new JRFWSAsync JMS Servers. For clarity and isolation, the following steps use individual persistent stores.

i. Create a persistence store for the new SOAJMSServer, for example, `SOAJMSFileStore_auto_n`. Specify the store's path, a directory on shared storage.

`ORACLE_BASE/admin/domain_name/cluster_name/jms/SOAJMSFileStore_auto_n`

j. Create a JMS Server for SOA, for example, `SOAJMSServer_auto_n`. Use `SOAJMSFileStore_auto_n` for this JMSServer. Target `SOAJMSServer_auto_n` to the new Managed Server, which is WLS_SOAn (migratable).

> **Note:**
>
> You can also assign `SOAJMSFileStore_n` as store for the new PS6 JMS Servers. For clarity and isolation, the following steps use individual persistent stores.

k. Update SubDeployment targets for SOA JMS Module to include the new SOA JMS Server. Expand the **Services** node, then expand the **Messaging** node. Choose **JMS Modules** from the Domain Structure window. Click **SOAJMSModule** (hyperlink in the **Names** column). In the Settings page, click the **SubDeployments** tab. In the subdeployment module, click the **SOAJMSServerXXXXXX** subdeployment and add `SOAJMSServer_n` to it Click **Save**.

> **Note:**
>
> A subdeployment module name is a random name in the form *COMPONENT*JMSServerXXXXXX. It comes from the Configuration Wizard JMS configuration for the first two Managed Servers, WLS_*COMPONENT*1 and WLS_*COMPONENT*2).

l. Update SubDeployment targets for UMSJMSSystemResource to include the new UMS JMS Server. Expand the **Services** node, then expand the **Messaging** node. Choose **JMS Modules** from the Domain Structure window. Click **UMSJMSSystemResource** (hyperlink in the Names column). In the Settings page, click the SubDeployments tab. In the subdeployment module, click the **UMSJMSServerXXXXXX** subdeployment and add `UMSJMSServer_n` to it. Click **Save**.

m. Update SubDeployment targets for OIMJMSModule to include the new OIM JMS Server. Expand the **Services** node, then expand **Messaging** node. Choose **JMS Modules** from the Domain Structure window. Click **OIMJMSModule** (hyperlink in the **Names** column). In the Settings page, click the SubDeployments tab. In the

subdeployment module, click **OIMJMSServerXXXXXX** and `OIMJMSServer_n` to it. Click **Save**.

n. Update SubDeployment targets for the JRFWSAsyncJmsModule to include the new JRFWSAsync JMS Server. Expand the **Services** node then expand the **Messaging** node. Choose **JMS Modules** from the Domain Structure window. Click **JRFWSAsyncJmsModule** (hyperlink in the Names column). In the Settings page, click the SubDeployments tab. Click the **JRFWSAsyncJMSServerXXXXXX** subdeployment and add `JRFWSAsyncJMSServer_`*n* to this subdeployment. Click **Save**

o. Update SubDeployment targets for BPM JMS Module to include the new BPM JMS Server. Expand the **Services** node, then expand the **Messaging** node. Choose **JMS Modules** from the Domain Structure window. Click **BPMJMSModule** (hyperlink in the **Names** column). In the Settings page, click the **SubDeployments** tab. In the subdeployment module, click the **BPMJMSServerXXXXXX** subdeployment and add `BPMJMSServer_`*n* to it. Click **Save**.

9. Run the pack command on SOAHOST1 to create a template pack. For example:

```
cd ORACLE_HOME/oracle_common/common/bin
./pack.sh -managed=true/
-domain=ORACLE_HOME/user_projects/domains/soadomain/
-template=soadomaintemplateScale.jar -
template_name='soa_domain_templateScale'
```

Run the following command on HOST1 to copy the template file created to HOST*n*:

```
scp soadomaintemplateScale.jar oracle@SOAHOSTN:/
ORACLE_BASE/product/fmw/soa/common/bin
```

Run the unpack command on HOSTn to unpack the template in the Managed Server domain directory. For example, for SOA:

```
ORACLE_HOME/oracle_common/common/bin
/unpack.sh /
-domain=ORACLE_HOME/user_projects/domains/soadomain/
-template=soadomaintemplateScale.jar
```

10. Configure the transaction persistent store for the new server. This should be a shared storage location visible from other nodes.

From the Administration Console, select **Server_name > Services** tab. Under Default Store, in Directory, enter the path to the folder where you want the default persistent store to store its data files.

11. Disable hostname verification for the new Managed Server; you must do this before starting/verifying the Managed Server. You can re-enable it after you configure server certificates for the communication between the Administration Server and Node Manager. If the source Managed Server (server you cloned the new one from) had already disabled hostname verification, these steps are not required. Hostname verification settings propagate to cloned servers.

To disable hostname verification:

a. Open the Administration Console.

b. Expand the **Environment** node in the Domain Structure window.

c. Click **Servers**.

d. Select WLS_SOAn in the **Names** column of the table. The Settings page for the server appears.

e. Click the SSL tab.

f. Click **Advanced**.

g. Set **Hostname Verification** to **None**.

h. Click **Save**.

12. Start Node Manager on the new node, as shown:

```
ORACLE_HOME/user_projects/domains/soadomain/bin/startNodeManager.sh
```

13. Start and test the new Managed Server from the Administration Console.

a. Shut down the existing Managed Server in the cluster.

b. Ensure that the newly created Managed Server is up.

c. Access the application on the newly created Managed Server to verify that it works. A login page appears for OIM. For SOA, a HTTP basic authorization opens.

Table 6-4 Managed Server Test URLs

Component	Managed Server Test URL
SOA	http://*vip*:*port*/soa-infra
OIM	http://*vip*:*port*/identity

14. Configure Server Migration for the new Managed Server.

> **Note:**
>
> Because this new node is using an existing shared storage installation, it is already using a Node Manager and environment configured for server migration that includes netmask, interface, wlsifconfig script superuser privileges. The floating IP for the new Managed Server is already in the new node.

To configure server migration:

a. Log into the Administration Console.

b. In the left pane, expand **Environment** and select **Servers**.

c. Select the server (represented as a hyperlink) for which you want to configure migration. The Settings page for that server appears.

d. Click the Migration tab.

e. In the Available field, in the Migration Configuration section, select machines to which to enable migration and click the right arrow.

> **Note:**
>
> Specify the least-loaded machine as the new server's migration target. Required capacity planning must be completed so that this node has the available resources to sustain an additional Managed Server.

f. Select the **Automatic Server Migration Enabled** option. This enables the Node Manager to start a failed server on the target node automatically.

g. Click **Save**.

h. Restart the Administration Server, Managed Servers, and Node Manager.

15. Test server migration for this new server from the node where you added it:

a. Stop the Managed Server.

Run `kill -9` *pid* on the PID of the Managed Server. Identify the PID of the node using, for example, `ps -ef | grep WLS_SOA`*n*.

b. Watch the Node Manager Console for a message indicating that the floating IP has been disabled.

c. Wait for the Node Manager to try a second restart of the new Managed Server. Node Manager waits for a fence period of 30 seconds before restarting.

d. After Node Manager restarts the server, stop it again. Node Manager should log a message that the server will not restart again locally.

16. Edit the OHS configuration file to add the new managed server(s). See Configuring Oracle HTTP Server to Recognize New Managed Servers.

- Configuring Oracle HTTP Server to Recognize New Managed Servers

Configuring Oracle HTTP Server to Recognize New Managed Servers

To complete scale up/scale out, you must edit the `oim.conf` file to add the new Managed Servers, then restart the Oracle HTTP Servers.

1. Go to the directory *OHS_DOMAIN_HOME*`/config/fmwconfig/components/OHS/instances/`*OHS_INSTANCE_NAME*.

2. Edit `mod_wl_ohs.conf` to add the new Managed Server to the WebLogicCluster directive. You must take this step for each URLs defined for OIM or SOA. Each product must have a separate `<Location>` section. Also, ports must refer to the Managed Servers. For example:

```
<Location /oim
    SetHandler weblogic-handler
    WebLogicCluster
 host1.example.com:14200,host2.example.com:14200
</Location>
```

3. Restart Oracle HTTP Server on WEBHOST1 and WEBHOST2:

```
WEBHOST1>OHS_DOMAIN_HOME/bin/stopComponent.sh OHS_Instance_NAME
WEBHOST1>OHS_DOMAIN_HOME/bin/startComponent.sh OHS_Instance_NAME

WEBHOST2>OHS_DOMAIN_HOME/bin/stopComponent.sh OHS_Instance_NAME
WEBHOST2>OHS_DOMAIN_HOME/bin/startComponent.sh OHS_Instance_NAME
```

> **Note:**
>
> If you are not using shared storage system (Oracle recommended), copy oim.conf to the other OHS servers.

> **Note:**
>
> See the General Parameters for WebLogic Server Plug-Ins in *Oracle Fusion Middleware Using Web Server 1.1 Plug-Ins with Oracle WebLogic Server* for additional parameters that can facilitate deployments.

Preparing for Shared Storage

Oracle Fusion Middleware allows you to configure multiple WebLogic Server domains from a single Oracle home. This allows you to install the Oracle home in a single location on a shared volume and reuse the Oracle home for multiple host installations.

If you plan to use shared storage in your environment, see Using Shared Storage in *High Availability Guide* for more information.

Deploying Oracle Identity and Access Management cluster with Unicast configuration

If multicast IP is disabled in deployment environment then you can deploy Oracle Identity and Access Management cluster with Unicast configuration.

In your deployment environment, if multicast IP is disabled because of security reasons or if you are using cloud Infrastructure for deployment, it is not feasible to deploy Oracle Identity and Access Management with the default configuration (multicast). It is not feasible because Oracle Identity and Access Management 12c uses EH cache, which depends on JavaGroup or JGroup library and supports multicast configuration as default for messages broadcasting.

To configure unicast configuration for EH cache, complete the following steps:

1. Get the list of host machines and available port to prepare the below JGroup configuration.

   Example:

   ```
   TCP(bind_port=7800;loopback=true):TCPPING(timeout=3000;initial_hosts=1.2.3.
   4[7800],1.2.3.5[7800];port_range=5;num_initial_members=2):pbcast.NAKACK(use_
   mcast_xmit=false;gc_lag=20;
   retransmit_timeout=1000):pbcast.GMS(print_local_addr=true;join_timeout=3000)
   ```

   In the example, you have servers: `1.2.3.4` and `1.2.3.5` and the available port on these machines is `7800`.

2. In the EM Console, expand **Identity and Access** > **OIM**.
3. Right-click on **oim(*version_number*)** and select **System MBean Browser**.

Where, *version_number* is the current version number of Oracle Identity and Access Management.

4. Expand **oracle.iam** > **Server:oim** > **Application:oim** > **XMLConfig** > **Config** > **XMLConfig.CacheConfig** > **Cache**.
5. In the right pane, **Attributes** tab, set the **Clustered** attribute value to `true`.
6. In the left pane, expand the **Cache** folder and select **XMLConfig.CacheConfig.XLCacheProvider** > **XLCacheProvider**.
7. In the right pane, **Attributes** tab, set the **MulicastConfig** attribute value to the equivalent JGroup string you identified in step 1.
8. Click **Apply**.
9. Restart all the Oracle Identity and Access Management managed servers.

7

Configuring High Availability for Oracle Access Manager Components

An introduction to Oracle Access Manager and description of how to design and deploy a high availability environment for Access Manager.

Access Manager provides a single authoritative source for all authentication and authorization services. See Introduction to Oracle Access Manager in *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*.

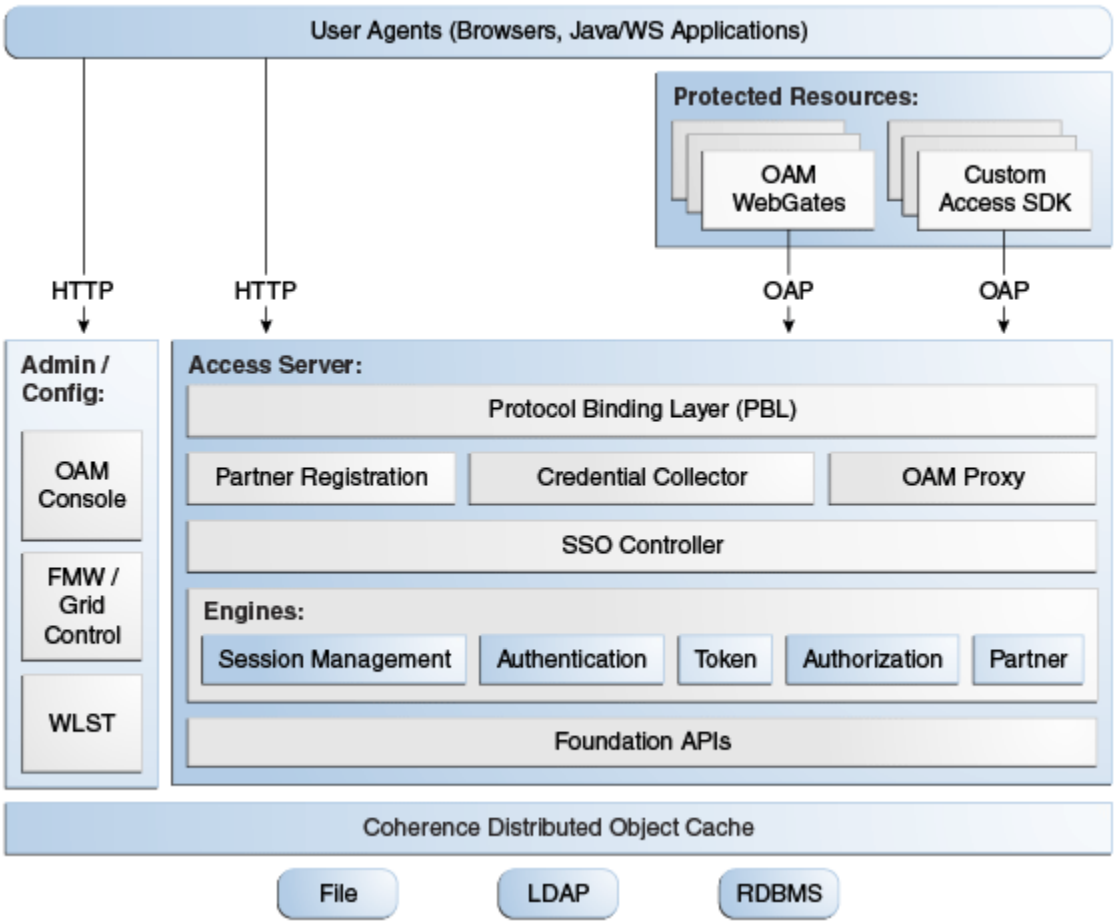
- Access Manager Component Architecture
  An introduction to primary Access Manager components and architecture.
- Access Manager High Availability Concepts
- High Availability Directory Structure Prerequisites
- Access Manager High Availability Configuration Steps
- Deploying Oracle Identity and Access Management cluster with Unicast configuration
  If multicast IP is disabled in deployment environment then you can deploy Oracle Identity and Access Management cluster with Unicast configuration.

Access Manager Component Architecture

An introduction to primary Access Manager components and architecture.

Figure 7-1 shows the Access Manager component architecture.

Figure 7-1 Access Manager Single Instance Architecture



Following are the components discussed in the Access Manager Single Instance Architecture:

- **User agents**: Include web browsers, Java applications, and Web services applications. User agents access the Access Server and administration and configuration tools using HTTP.
- **Protected resources**: Application or web page to which access is restricted. WebGates or Custom Agents control access to protected resources.
- **Administration and configuration tools**: Administer and configure Access Manager with Oracle Access Management Console, Oracle Enterprise Manager Fusion Middleware Control and Oracle Enterprise Manager Grid Control, and WebLogic Scripting Tool (WLST).
- **Access Server**: Includes Credential Collector and OAM Proxy components.
- Access Manager Component Characteristics
  A typical Access Manager deployment consists of system entities such as user agents, protected resources, and access server.
- Access Manager Configuration Artifacts
- Access Manager External Dependencies

Access Manager Component Characteristics

A typical Access Manager deployment consists of system entities such as user agents, protected resources, and access server.

A list of system entities and the characteristics required for an Access Manager deployment:

- **Access Manager Agents** - Access Server extensions that ensure access is controlled according to policies that Access Server manages. Agents require the Access Server component to perform their functions. If Access Server is unavailable, access to protected servers is denied; users are locked out of the system.
- **Protected Resources** (partnered applications) - Applications that Access Manager protects. Access to these resources depends on access control policies in Access Manager and enforced by Access Manager agents deployed in the protected resource's access path.
- **Access Server** - Server side component. Provides core runtime access management services.
- **JMX Mbeans** - Runtime Mbeans are packaged as part of the Access Server package. Config Mbeans are packaged as standalone WAR files.
- **WebLogic 12*c* SSPI providers** consist of Java classes that implement the SSPI interface along with Access Java Access JDK. AccessGates are built using pure Java Access JDK.
- **Oracle Access Management Console** - Application that hosts Administration Console and provides services to manage Access Manager deployment.
- **WebLogic Scripting Tool** - Java classes included in Access Server package. Limited administration of Access Manager deployment is supported via the command line.
- **Fusion Middleware Control and Enterprise Manager Grid Control** - Access Manager integrates with Enterprise Manager Grid Control to show performance metrics and deployment topology.
- **Access Manager Proxy** - Custom version of Apache MINA server. Includes MessageDrivenBeans and ResourceAdapters in addition to Java Server classes.
- **Data Repositories** - Access Manager handles different types of information including Identity, Policy, Partner, Session and Transient data:
  - LDAP for Identity data
  - Files for Configuration and Partner data
  - Policy data will be stored in files or in an RDBMS
- Oracle **Access Manager WebGates** are C-based agents that are intended to be deployed in web servers.
- **Oracle Single Sign-On Apache** modules are C-based agents that are intended to be deployed in Oracle HTTP Server web servers.

Access Manager Configuration Artifacts

Access Manager configuration artifacts include:

Table 7-1 Access Manager Configuration Artifacts

Configuration Artifact	Description
DOMAIN_HOME/config/fmwconfig/oam-config.xml	Configuration file which contains instance specific information.
DOMAIN_HOME/config/fmwconfig/oam-policy.xml	Policy store information.
DOMAIN_HOME/config/fmwconfig/.oamkeystore	Stores symmetric and asymmetric keys.
DOMAIN_HOME/config/fmwconfig/component_events.xml	Used for audit definition.
DOMAIN_HOME/config/fmwconfig/jazn-data.xml	Administration Console permissions
DOMAIN_HOME/config/fmwconfig/servers/*instanceName*/logging.xml	Logging configuration. Do not edit this file manually.
DOMAIN_HOME/config/fmwconfig/servers/*instanceName*/dms_config.xml	Tracing logging. Do not edit this file manually.
DOMAIN_HOME/config/fmwconfig/cwallet.sso	Stores passwords that OAM uses to connect to identity stores, database, and other entities. This is not for end user passwords.
DOMAIN_HOME/output	Stores agent configuration files.

Access Manager External Dependencies

The following table describes Access Manager external runtime dependencies.

Table 7-2 Access Manager External Dependencies

Dependency	Description
LDAP based Identity Store	• User Identity Repository • LDAP access abstracted by User/Role API. Access Manager always connects to one Identity store: a physical server or a load balancer IP. If the primary down, Access Manager reconnects and expects the load balancer to connect it to the secondary.
OCSP Responder Service	Real-time X.509 Certification Validation
RDBMS Policy Store	• Policy (Authentication and Authorization) Repository • RDBMS access abstracted by the OAM policy engine
Oracle Identity Manager Policy Store (when Oracle Identity Manager-based password management is enabled)	LDAP Repository containing Oblix Schema elements that are used to store Configuration, Metadata, and so on
Identity Federation	Dependency when Identity Federation Authentication Scheme is selected
OCSP Responder Service	Real-time X.509 Certification Validation

• Access Manager Log File Location

Access Manager Log File Location

You deploy Access Manager on WebLogic Server. Log messages go to the server log file of the WebLogic Server that you deploy it on. The default server log location is:

```
Domain_HOME/servers/serverName/logs/ serverName-diagnostic.log
```

Access Manager High Availability Concepts

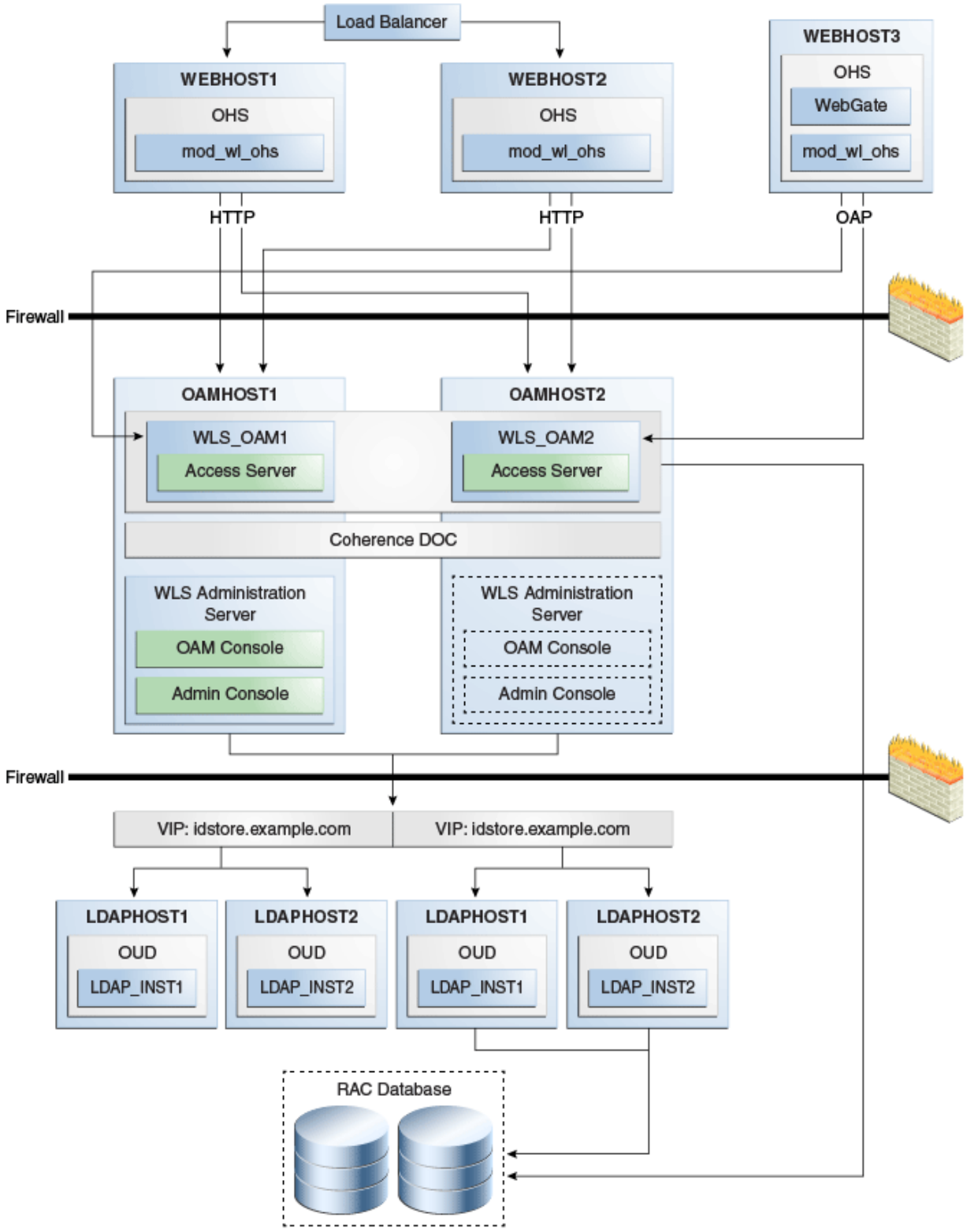
This following sections provide conceptual information about using Access Manager in a high availability two-node cluster.

- Access Manager High Availability Architecture
- Protection from Failures and Expected Behaviors

Access Manager High Availability Architecture

Figure 7-2 shows an Access Manager high availability architecture:

Figure 7-2 Access Manager High Availability Architecture



In Figure 7-2, the hardware load balancer receives incoming authentication requests and routes them to WEBHOST1 or WEBHOST2 in the web tier. These hosts have

Oracle HTTP Server installed. Oracle HTTP Server then forwards requests on to the WebLogic managed servers using the WebLogic plugin `mod_wl_ohs.conf`. See Oracle HTTP Server Configuration.

The load balancing router should use session stickiness for HTTP traffic only. OAP traffic does not use a load balancing router, so session stickiness is not required for OAP traffic.

Applications that other Oracle HTTP Servers access, that in turn have resources with restricted access, must have a WebGate and a custom agent configured. The WebGate on WEBHOST3 communicates with the Access Servers on OAMHOST1 and OAMHOST2 in the application tier using OAP. WEBHOST3 is an application web server, and for authentication, HTTP redirect routes requests to the load balancer and WEBHOST1 and WEBHOST2. For a high availability deployment, you can configure another host (for example, WEBHOST4) with the same components as WEBHOST3.

OAMHOST1 and OAMHOST2 deploy managed servers which host the Oracle Access Server application. These managed servers are configured in a cluster which enables the Access Servers to work in an active-active manner.

The Administration Server runs on OAMHOST1 and deploys the WebLogic Administration Console, Oracle Enterprise Manager Fusion Middleware Control, and the Oracle Access Management Console.

In the directory tier, the virtual IP `idstore.example.com` routes the IDstore requests to LDAPHOST1 and LDAPHOST2, which comprise an active-active IDStore cluster. For example the virtual IP `oud.example.com` is set up to route Oracle Unified Directory requests to OUDHOST1 and OUDHOST2, which comprise an active-active Oracle Unified Directory cluster.

An Oracle RAC database provides high availability in the data tier. The Oracle RAC database is configured in a JDBC multi data source or GridLink data source to protect the instance from Oracle RAC node failure.

In Access Manager 12*c*, only one Access Manager cluster is supported per WebLogic Server domain. Access Manager clusters cannot span WebLogic Server domains.

A single instance Access Manager deployment satisfies the following high availability requirements:

- Load handling
- External connection management and monitoring
- Recovery
- Fault containment
- Fault diagnostics
- Administration Server offline

A multiple instance Access Manager deployment satisfies the following additional high availability requirements:

- Redundancy
- Client connection failover/continuity
- Client load balancing
- State management

Oracle recommends using an external load balancing router for inbound HTTP connections. Outbound external connections to LDAP Servers (or OAM policy engine PDP/PIP) are load balanced with support for connection failover. Therefore, a load balancer is not required. Access Manager agents, typically WebGates, can load balance connections across multiple Access Servers.

Access Manager agents open persistent TCP connections to the Access Servers. This requires firewall connection timeouts to be sufficiently large to avoid premature termination of TCP connections.

The Access Server and Access Manager Administration Console interface with the OAM policy engine for policy evaluation and management. The OAM policy engine internally depends on a database as the policy repository. The database interactions are encapsulated within the OAM policy engine, with only the connectivity configuration information managed by Access Manager. The high availability characteristics of the interaction between Access Manager and the OAM policy engine are:

- The database connection information is configured in the Access Manager configuration file synchronized among the Access Manager instances.
- Database communication is managed within the OAM policy engine, and generally decoupled from Access Manager and OAM policy engine interactions. The very first startup of an OAM server instance will fail, however, if the database is unreachable. An OAM policy engine bootstrap failure is treated as fatal by Access Manager, and the startup operation is aborted.
- Access Manager policy management interfaces (in the Oracle Access Management Console and the CLI tool) fail if the database is unreachable, as seen by the OAM policy engine management service interfaces. The operation may be retried at a later point in time, but no automated retry is provided for management operations.
- Following a successful policy modification in the database repository, the OAM policy engine layer in the OAM server runtimes retrieves and activates the changes within a configurable OAM policy engine database poll interval (configured through Access Manager configuration). A positive acknowledgement of a policy change must be received from each OAM server runtime, otherwise the policy change cannot be considered successfully activated. The administrator can use the Oracle Access Management Console to remove any Access Manager instance with a policy activation failure from service.

Protection from Failures and Expected Behaviors

The WebLogic Server infrastructure protects the Identity Management Service Infrastructure system from all process failures. These features protect an Access Manager high availability configuration from failure

- Back channel OAP bindings use a primary/secondary model for failover. Front Channel HTTP bindings use a load balancing router for failover.
- If an Access Server fails, a WebGate with a persistent connection to that server waits for the connection to timeout, then it switches over to the secondary (backup) Access Server. Outstanding requests fail over to the secondary server.
- Access Manager Access Servers support a heartbeat check. Also, the WebLogic Node Manager on the Managed Server can monitor the application and restart it.

- If a WebLogic Server node fails, external connection failover is based on the configuration, the retry timeout, and the number of retries. Access Manager Agent-Access Server failover is based on a timeout.
- If the load balancing router or proxy server detects a WebLogic Server node failure, subsequent client connections route to the active instance, which picks up the session state and carries on with processing.
- When the lifetime of a connection expires, pending requests complete before the connection terminates. The connection object returns to the pool.
- When it receives an exception from another service, Access Manager retries external connection requests. You can configure the number of retries.
- WebLogic Server Crash
- Node Failure
- Database Failure

WebLogic Server Crash

If a Managed Server fails, Node Manager attempts to restart it locally

Ongoing requests from Oracle HTTP Server timeout and new requests are directed to the other Managed Server. After the server's restart completes on the failed node, Oracle HTTP Server resumes routing any incoming requests to the server.

> **Note:**
>
> Access Manager servers support a heartbeat check to determine if the access server can service its requests. It checks:
>
> - Whether the LDAP store can be accessed
> - Whether the policy store can be accessed
>
> If the heartbeat succeeds, the Access Server can service requests and requests are sent to it. If the heartbeat fails, requests do not route to the Access Server.

Node Failure

Node failures are treated in the same way as WebLogic Server fails.

Database Failure

Multi data sources protect Access Manager service Infrastructure against failures. When an Oracle RAC database instance fails, connections are reestablished with available database instances. The multi data source enables you to configure connections to multiple instances in an Oracle RAC database.

For more on multi data source configuration, see Section 4.1.3, "Using Multi Data Sources with Oracle RAC".

High Availability Directory Structure Prerequisites

A high availability deployment requires product installations and files to reside in specific directories. A standard directory structure facilitates configuration across nodes and product integration.

The following table describes high availability directory structure prerequisites.

Table 7-3 Directory Structure Prerequisites

Directory	Requirements
ORACLE_HOME	**Each product must have its own ORACLE_HOME**. For example, OAM and OIM must go in separate *ORACLE_HOME* locations. **ORACLE_HOME contents must be identical across all nodes.** Across all nodes, *ORACLE_HOME* must: • Reside in the file system at the same path • Contain identical products • Contain identical versions of those products • Have identical *ORACLE_HOME* names • Have identical patches installed
DOMAIN_HOME and *APPLICATION_DIRECTORY*	These directories must have the same path on all nodes. Put these directories in a separate file system location from *ORACLE_HOME*; do not put these directories in the *ORACLE_HOME*/`user_projects` directory
`wlsserver_10.`*n*	Each OAM and OIM installation requires its own, separate WebLogic Server installation.

You have three options to set up the high availability directory structure:

- Use shared storage to store *ORACLE_HOME* directories. Oracle recommends this option. Use a NFS exported by a NAS, or a cluster file system pointing to a SAN/NAS.
- Use local storage and run all installation, upgrade, and patching procedures on one node, then replicate to other nodes (using rsync, for example.)
- Use local storage and repeat all installation and patch procedures on each node.

Access Manager High Availability Configuration Steps

This section provides high-level instructions to set up a high availability deployment for Access Manager. This deployment includes two Oracle HTTP Servers, which distribute requests to two OAM servers. These OAM servers interact with an Oracle Real Application Clusters (Oracle RAC) database and, optionally, an external LDAP store. If any single component fails, the remaining components continue to function.

See Using Dynamic Clusters.

- Access Manager Configuration Prerequisites
- Running the Repository Creation Utility to Create the Database Schemas
- Installing Oracle WebLogic Server
- Installing and Configuring the Access Manager Application Tier

Access Manager Configuration Prerequisites

Before you configure Access Manager for high availability, you must:

- Install Oracle WebLogic Server on OAMHOST1 and OAMHOST2. See Installing Oracle WebLogic Server.
- Install the Oracle Identity Management executables on OAMHOST1 and OAMHOST2. See the Installing and Configuring the Access Manager Application Tier.
- Run the Repository Creation Utility to create the Access Manager schemas in a database. See Running the Repository Creation Utility to Create the Database Schemas.
- Ensure that a highly available LDAP implementation is available.

For example,

- Install the Infrastructure jar, `jdk8/bin/java -jar fmw_12.2.1.4.0_infrastructure.jar` and change the default installation directory path manually from `/tmp/Middleware/ORACLE_HOME` to `/tmp/Middleware/`
- Install IDM jar, `jdk8/bin/java -jar fmw_12.2.1.4.0_idm.jar` and choose `/tmp/Middleware/` as the installation directory.
- Run RCU located at `/tmp/Middleware/oracle_common/bin/rcu`

Running the Repository Creation Utility to Create the Database Schemas

The schemas you create depend on the products you want to install and configure. See Starting the Repository Creation Utility to run RCU.

For more information, see *Planning an Installation of Oracle Fusion Middleware* and *Creating Schemas with the Repository Creation Utility*.

Installing Oracle WebLogic Server

To install Oracle WebLogic Server, see *Installing and Configuring Oracle WebLogic Server and Coherence*.

> **Note:**
>
> On 64-bit platforms, when you install Oracle WebLogic Server using the generic jar file, JDK is not installed with Oracle WebLogic Server. You must install JDK separately, before installing Oracle WebLogic Server.

Installing and Configuring the Access Manager Application Tier

See Installing and Configuring the Oracle Access Management Software in *Installing and Configuring Oracle Identity and Access Management*.

Creating boot.properties for the Administration Server on OAMHOST1

The `boot.properties` file enables the Administration Server to start without prompting for the administrator's username and password.

To create the `boot.properties` file:

1. On OAMHOST1, go to:

   ORACLE_HOME`/user_projects/domains/`*domainName*`/servers/AdminServer/security`

   For example:

   ```
   cd /u01/app/oracle/product/fmw/user_projects/domains/IDMDomain/servers/
   AdminServer/security
   ```

2. Use a text editor to create a file called `boot.properties` under the `security` directory. Enter the following lines in the file:

   ```
   username=adminUser
   password=adminUserPassword
   ```

   > **Note:**
   >
   > When you start Administration Server, username and password entries in the file get encrypted. For security reasons, minimize the time the entries in the file are left unencrypted. After you edit the file, start the server as soon as possible to encrypt the entries.

3. Stop the Administration Server if it is running.

   See Starting and Stopping Oracle Fusion Middleware in *Administering Oracle Fusion Middleware* to start and stop WebLogic Servers.

4. Start Node Manager by using the following commands:

   ```
   cd WL_HOME/server/bin ./startNodeManager.sh
   ```

5. Start the Administration Server on OAMHOST1 with the `startWebLogic.sh` script in the *ORACLE_HOME*`/user_projects/domains/`*domainName*`/bin` directory.

6. Validate that changes are successful. Open a browser and log into these consoles using the `weblogic` user credentials:

- WebLogic Server Administration Console at:

  `http://oamhost1.example.com:7001/console`

- Oracle Enterprise Manager Fusion Middleware Control at:

  `http://oamhost1.example.com:7001/em`

Starting OAMHOST1

The following sections describe the steps for starting OAMHOST1.

- Start Node Manager
- Start Access Manager on OAMHOST1

Start Node Manager

- Start Node Manager by issuing the following command:

  ```
  OAMHOST1>ORACLE_HOME/user_projects/domains/domainName/bin/startNodeManager.sh
  ```

Start Access Manager on OAMHOST1

To start Access Manager on OAMHOST1, follow these steps:

1. Log into the WebLogic Administration Console using this URL using WebLogic administrator credentials:

   `http://oamhost1.example.com:7001/console`

2. Start the WLS_OAM1 Managed Server using the WebLogic Server Administration Console, as follows:
   a. Expand the **Environment** node in the **Domain Structure** tree on the left.
   b. Click **Servers**.
   c. On the Summary of Servers page, open the **Control** tab.
   d. Select **WLS_OAM1**, and then click **Start**.
   e. Click **YES** to confirm that you want to start the server.
   f. Then select **OAM_POLICY_MGR1**, and then click **Start**.
   g. Click **YES** to confirm that you want to start the server.

Validating OAMHOST1

Validate the implementation by connecting to the OAM server:

```
http://OAMHOST1.example.com:14150/access
http://OAMHOST1.example.com:14100/oam/server/logout
```

The implementation is valid if an OAM logout successful page opens.

Configuring OAM on OAMHOST2

After configuration succeeds on OAMHOST1, propagate it to OAMHOST2. Pack the domain using the `pack` script on OAMHOST1 and unpack it with the `unpack` script on OAMHOST2.

Both scripts reside in the *ORACLE_HOME*/oracle_common/common/bin directory.

On OAMHOST1, enter:

```
pack.sh -domain=$ORACLE_HOME/user_projects/domains/IDM_Domain \
    -template=/tmp/idm_domain.jar -template_name=OAM Domain -managed=true
```

This creates a file called idm_domain.jar in the /tmp directory. Copy this file to OAMHOST2.

On OAMHOST2, enter:

```
unpack.sh -domain=$ORACLE_HOME/user_projects/domains/IDM_Domain \
    -template=/tmp/idm_domain.jar
```

Starting OAMHOST2

This following sections describe the steps for starting OAMHOST2. Steps include the following:

- Create the Node Manager Properties File on OAMHOST2
- Start Node Manager
- Start Access Manager on OAMHOST2

Create the Node Manager Properties File on OAMHOST2

Before you can start managed servers from the console, you must create a Node Manager property file. Run the script setNMProps.sh, which is located in the *ORACLE_HOME*/oracle_common/common/bin directory. For example:

```
OAMHOST1> $ORACLE_HOME/oracle_common/common/bin/setNMProps.sh
```

Start Node Manager

Start Node Manager by issuing the following command:

```
OAMHOST2>ORACLE_HOME/user_projects/domains/domainName/bin/startNodeManager.sh
```

Start Access Manager on OAMHOST2

To start Access Manager on OAMHOST2:

1. Log into the WebLogic Administration Console using this URL:

   ```
   http://OAMHOST1.example.com:7001/console
   ```

2. Supply the WebLogic administrator username and password.
3. Select **Environment - Servers** from the **Domain Structure** menu.
4. Click the Control tab.
5. Click the server **WLS_OAM2**.
6. Click **Start**.
7. Click **OK** to confirm that you want to start the server.

Validating OAMHOST2

Validate the implementation by connecting to the OAM server:

```
http://OAMHOST2.example.com:14150/access
http://OAMHOST2.example.com:14100/oam/server/logout
```

The implementation is valid if an OAM logout successful page opens.

Configuring Access Manager to Work with Oracle HTTP Server

Complete the subsequent procedures to configure Access Manager to work with Oracle HTTP Server.

- Update Oracle HTTP Server Configuration
- Restart Oracle HTTP Server
- Make OAM Server Aware of the Load Balancer

Update Oracle HTTP Server Configuration

On WEBHOST1 and WEBHOST2, create a file named `oam.conf` in this directory:

```
OHSDomain/config/fmwconfig/components/OHS/<instancename>/moduleconf/
```

Create the file and add the following lines:

```
NameVirtualHost *:7777
<VirtualHost *:7777>

    ServerName login.example.com:7777
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit

    <Location /oam>
        SetHandler weblogic-handler
        Debug ON
        WLLogFile /tmp/weblogic.log
        WLProxySSL ON
        WLProxySSLPassThrough ON
        WLCookieName OAM_JSESSIONID
        WebLogicCluster OAMHOST1.example.com:14100,OAMHOST2.example.com:14100
    </Location>

    <Location /oamfed>
        SetHandler weblogic-handler
        Debug ON
        WLLogFile /tmp/weblogic.log
        WLProxySSL ON
        WLProxySSLPassThrough ON
        WLCookieName OAM_JSESSIONID
        WebLogicCluster OAMHOST1.example.com:14100,OAMHOST2.example.com:14100

    </Location>

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

```
        WLLogFile /tmp/weblogic.log
        WLProxySSL ON
        WLProxySSLPassThrough ON
        WLCookieName OAM_JSESSIONID
        WebLogicCluster OAMHOST1.example.com:14100,OAMHOST2.example.com:14100

    </Location>

    <Location /access>
        SetHandler weblogic-handler
        Debug ON
        WLLogFile /tmp/weblogic.log
        WLProxySSL ON
        WLProxySSLPassThrough ON
        WLCookieName OAM_JSESSIONID
        WebLogicCluster amahost1.example.com:14150,amahost2.example.com:14150


    <Location /oamsso>
        SetHandler weblogic-handler
        Debug ON
        WLLogFile /tmp/weblogic.log
        WLProxySSL ON
        WLProxySSLPassThrough ON
        WLCookieName OAM_JSESSIONID
        WebLogicCluster oam_policy_mgr1.example.com:14100,oam_policy_
         mgr2.example.com:14100

    </Location>

</VirtualHost>
```

Restart Oracle HTTP Server

Restart the Oracle HTTP Server on WEBHOST1:

```
OHSDomain/bin/stopComponent.sh ohs1
OHSDomain/bin/startComponent.sh ohs1
```

Restart the Oracle HTTP Server on WEBHOST2:

```
OHSDomain/bin/stopComponent.sh ohs2
OHSDomain/bin/startComponent.sh ohs2
```

Make OAM Server Aware of the Load Balancer

By default, Access Manager sends requests to the login page on the local server. In a high availability deployment, you must change this setup so that login page requests go to the load balancer.

To make Access Manager aware of the load balancer:

1. Log into the Oracle Access Management Console at this URL as the `weblogic` user:

   ```
   http://OAMHOST1.example.com:7001/oamconsole
   ```

2. Click on the **Configuration** tab.
3. Click the **Access Manager Settings** link.

4. Enter the following information:
   - OAM Server Host: login.example.com
   - OAM Server Port: 7777
   - OAM Server Protocol: https
5. Click **Apply**.
6. Restart managed servers WLS_OAM1 and WLS_OAM2.

Configuring Access Manager to use an External LDAP Store

By default, Access Manager uses its own in built-in LDAP server. In a highly available environment, Oracle recommends an external LDAP directory as the directory store.

> **Note:**
>
> Oracle recommends that you back up the environment and LDAP store before following this procedure.

- Extending Directory Schema for Access Manager
- Creating Users and Groups in LDAP
- Creating a User Identity Store
- Setting LDAP to System and Default Store
- Setting Authentication to Use External LDAP
- Adding LDAP Groups to WebLogic Administrators
  Access Manager requires access to MBeans stored within the administration server. In order for LDAP users to be able to log in to the WebLogic console and Fusion Middleware control, they must be assigned the WebLogic Administration rights. In order for Access Manager to invoke these Mbeans, users in the IAMAdministrators group must have WebLogic Administration rights.

Extending Directory Schema for Access Manager

Pre-configuring the Identity Store extends the schema in the backend directory regardless of directory type.

To extend the directory schema for Access Manager, perform these steps on OAMHOST1:

1. Set the Environment Variables: `JAVA_HOME`, `IDM_HOME` and `ORACLE_HOME`.

   Set `IDM_HOME` to `IDM_ORACLE_HOME`

   Set `ORACLE_HOME` to `IAM_ORACLE_HOME`
2. Create a properties file `extend.props` that contains the following:

```
IDSTORE_HOST : idstore.example.com

IDSTORE_PORT : 389

IDSTORE_BINDDN : cn=orcladmin
```

```
IDSTORE_USERNAMEATTRIBUTE: cn

IDSTORE_LOGINATTRIBUTE: uid

IDSTORE_USERSEARCHBASE:cn=Users,dc=example,dc=com

IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=us,dc=oracle,dc=com

IDSTORE_SEARCHBASE: dc=example,dc=com

IDSTORE_SYSTEMIDBASE: cn=systemids,dc=example,dc=com
```

Where:

- `IDSTORE_HOST` and `IDSTORE_PORT` are, respectively, the host and port of your Identity Store directory. Specify the back-end directory here rather than OUD.)
- `IDSTORE_BINDDN` Administrative user in the Identity Store Directory
- `IDSTORE_USERSEARCHBASE` Location in your Identity Store where users are placed.
- `IDSTORE_GROUPSEARCHBASE` Location in your Identity Store where groups are placed.
- `IDSTORE_SEARCHBASE` Location in the directory where Users and Groups are stored.
- `IDSTORE_SYSTEMIDBASE` Location in your directory where the Oracle Identity Manager reconciliation users are placed.
- `IDSTORE_SYSTEMIDBASE` Location of a container in the directory where you can place users when you do not want them in the main user container. This happens rarely. For example, if Oracle Identity Manager reconciliation user which is also used for the bind DN user in Oracle Virtual Directory adapters.

3. Configure the Identity Store using the command `idmConfigTool`, located at `IAM_ORACLE_HOME/idmtools/bin`.

   The command syntax is:

   ```
   idmConfigTool.sh -preConfigIDStore input_file=configfile
   ```

   For example:

   ```
   idmConfigTool.sh -preConfigIDStore input_file=extend.props
   ```

   The system prompts you for the account password with which you are connecting to the Identity Store.

   Sample command output:

   ```
   Enter ID Store Bind DN password :
   Apr 5, 2011 3:39:25 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
   INFO: -> LOADING:


   /u01/app/oracle/product/fmw/IAM/idmtools/templates/oid/
   idm_idstore_groups_template.ldif
   Apr 5, 2011 3:39:25 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
   INFO: -> LOADING:
   ```

```
/u01/app/oracle/product/fmw/IAM/idmtools/templates/oid/
idm_idstore_groups_acl_template.ldif
Apr 5, 2011 3:39:25 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING:


/u01/app/oracle/product/fmw/IAM/idmtools/templates/oid/systemid_pwdpolicy.ldif
Apr 5, 2011 3:39:25 AM oracle.ldap.util.LDIFLoader loadOneLdifFileINFO: -> LOADING:


/u01/app/oracle/product/fmw/IAM/idmtools/templates/oid/idstore_tuning.ldifApr 5,
2011 3:39:25 AM oracle.ldap.util.LDIFLoader loadOneLdifFileINFO: -> LOADING:


/u01/app/oracle/product/fmw/IAM/idmtools/templates/oid/oid_schema_extn.ldif
The tool has completed its operation. Details have been logged to automation.log
```

4. Check the log file for errors and warnings and correct them.

Creating Users and Groups in LDAP

To add users that Access Manager requires to the Identity Store, follow these steps:

1. Set the Environment Variables `JAVA_HOME`, `IDM_HOME`, and `ORACLE_HOME`.
   - Set `IDM_HOME` to `IDM_ORACLE_HOME`.
   - Set `ORACLE_HOME` to `IAM_ORACLE_HOME`.
2. Create a properties file `oam.props` that contains the following parameters shown in the following example:

```
IDSTORE_HOST: host.example.com
IDSTORE_PORT: 9389
IDSTORE_BINDDN: cn=directory manager
IDSTORE_PASSWD: secret12
IDSTORE_USERNAMEATTRIBUTE: cn
IDSTORE_LOGINATTRIBUTE: uid
IDSTORE_USERSEARCHBASE: ou=people,dc=example,dc=com
IDSTORE_GROUPSEARCHBASE: ou=groups,dc=example,dc=com
IDSTORE_SEARCHBASE: dc=example,dc=com
IDSTORE_SYSTEMIDBASE: cn=systemids,dc=example,dc=com
IDSTORE_OAMSOFTWAREUSER: oamSoftwareUser
IDSTORE_PWD_OAMSOFTWAREUSER: example_pwd
IDSTORE_OAMADMINUSER: oamAdminUser
IDSTORE_PWD_OAMADMINUSER: example_pwd
IDSTORE_PWD_OBLIXANONYMOUSUSER: example_pwd
IDSTORE_PWD_ANONYMOUSUSER: example_pwd
OAM11G_IDSTORE_ROLE_SECURITY_ADMIN: OAMAdministrators
POLICYSTORE_SHARES_IDSTORE: true
```

3. Configure the Identity Store using the command `idmConfigTool` which is located at `IAM_ORACLE_HOME/idmtools/bin`.

The command syntax is as shown in the following example:

```
$ORACLE_HOME/idmtools/bin/idmConfigTool.sh -prepareIDStore mode=OAM
input_file=prepareIDStore.properties log_level=ALL
log_file=log_idstore1.out dump_params=true
```

After the command runs, the system prompts you to enter the password for the account with which you are connecting to the ID Store.

4. Check the log file for any errors or warnings and correct them.

Creating a User Identity Store

To create a user identity store:

1. Go to the Oracle Access Management Console at the URL:

   ```
   http://adminvhn.example.com:7001/oamconsole
   ```

2. Log in using the WebLogic administration user.
3. Select **Configuration** tab and click **User Identity Stores**.
4. Under OAM ID Stores, click **Create**. Enter the following information:
   - **Store Name**: LDAP_DIR
   - **Store Type**: OUD
   - **Description**: Enter a description of the Directory Store
   - **Enable SSL**: Select this if you communicate with your directory over SSL
   - **Location**: Enter the location, for example oud.example.com:389
   - **Bind DN**: Enter the user permitted to search the LDAP store. For example, cn=orcladmin
   - **Password**: Enter the oracleadmin password
   - **User Name Attribute**: For example: uid
   - **User Search Base**: Enter the location of users in the LDAP store. For example, cn=Users,dc=example,dc=com
   - **Group Name Attribute**: For example: orclguid
   - **Group Search Base**: Enter the location of groups in the LDAP store. For example, cn=Groups,dc=example,dc=com
   - **OAM Administrator Role**: OAMAdministrators
5. Click **Apply**.
6. Click **Test Connection** to validate the connection to the LDAP server.

Setting LDAP to System and Default Store

After you define the LDAP identity store, you must set it as the primary authentication store. Follow these steps in the Oracle Access Management Console:

1. From the **Configuration** tab, click **User Identity Stores**.
2. Select **LDAP_DIR** as Default Store.

3. Select **LDAP_DIR** as System Store.
4. Click the Add **[+]** icon in **Access System Administrators**.
5. Enter **OAM*** in the search name field and click **Search**.
6. Select **OAMAdministrators** from the search results and click **Add Selected**.
7. Click **Apply**.
8. In the Validate System Administrator window, enter the username and password of the OAM administrator, for example, oamadmin.
9. Click **Validate**.
10. Test the connection by clicking **Test Connection**.

Setting Authentication to Use External LDAP

By default, Access Manager uses the integrated LDAP store for user validation. You must update the LDAP authentication module so that it can validate users against the new external LDAP store.

To update the LDAP authentication module to use external LDAP:

1. Under **Application Security** tab, select **Authentication Modules** and click **Search**.
2. Click **LDAP**.
3. Select **Open** from the **Actions** menu.
4. Set **User Identity Store** to `LDAP_DIR`.
5. Click **Apply**.
6. Restart the Managed Servers Admin Server, WLS_OAM1 and WLS_OAM2.

Adding LDAP Groups to WebLogic Administrators

Access Manager requires access to MBeans stored within the administration server. In order for LDAP users to be able to log in to the WebLogic console and Fusion Middleware control, they must be assigned the WebLogic Administration rights. In order for Access Manager to invoke these Mbeans, users in the IAMAdministrators group must have WebLogic Administration rights.

When Single Sign-on is implemented, provide the LDAP group IDM Administrators with WebLogic administration rights, so that you can log in using one of these accounts and perform WebLogic administrative actions.

To add the LDAP Groups `IAMAdministrators` and `WLSAdmins` to the WebLogic Administrators:

1. Log in to the WebLogic Administration Server Console.
2. In the left pane of the console, click **Security Realms**.
3. On the Summary of Security Realms page, click **myrealm** under the Realms table.
4. On the Settings page for **myrealm**, click the **Roles & Policies** tab.
5. On the Realm Roles page, expand the **Global Roles** entry under the Roles table.
6. Click the **Roles** link to go to the Global Roles page.
7. On the Global Roles page, click the **Admin** role to go to the Edit Global Roles page.

8. On the Edit Global Roles page, under the **Role Conditions** table, click the **Add Conditions** button.
9. On the Choose a Predicate page, select **Group** from the drop down list for predicates and click **Next**.
10. On the Edit Arguments Page, Specify **IAMAdministrators** in the **Group Argument** field and click **Add**.
11. Repeat for the Group **WLSAdmins**.
12. Click **Finish** to return to the Edit Global Roles page.
13. The **Role Conditions** table now shows the groups **IAMAdministrators and WLSAdmins** as role conditions.
14. Click **Save** to finish adding the Admin role to the OAMAdministrators and IDM Administrators Groups.

Validating the Access Manager Configuration

Validate the configuration by logging into the Oracle Access Management Console at `http://OAMHOST1.example.com:7001/oamconsole` as `oamadmin`.

See Adding LDAP Groups to WebLogic Administrators

Scaling Up Access Manager Topology

You *scale up* to add a new Access Manager managed server to a node already running one or more server instances.

- Scaling Up Access Manager
- Registering the New Managed Server
- Configuring WebGate with the New OAM Managed Server

Scaling Up Access Manager

To scale up OAM:

1. Log in to the Administration Console at `http://hostname.example.com:7001/console`. From the Domain Structure window, expand the **Environment** node and then **Servers**.
2. In the Change Center, click **Lock & Edit**.
3. Select a server on the host you want to extend, for example: `WLS_OAM1`.
4. Click **Clone**.
5. Enter the following information:
   - **Server Name**: A new name for the server, for example: `WLS_OAM3`.
   - **Server Listen Address**: The name of the host on which the managed server will run.
   - **Server Listen Port**: The port the new managed server will use, this port must be unique within the host.
6. Click **OK**.

7. Click on the newly created server **WLS_OAM3**
8. Set the SSL listen port. This should be unique on the host that the managed server will run on.

> **Note:**
>
> Enable the SSL listen port 14101.

9. Click **Save**.
10. Disable hostname verification for the new managed server. You must do this before you start and verify the `WLS_OAM3` Managed Server. You can re-enable it after you configure server certificates for the communication between the Administration Server and Node Manager in `OAMHOST`*`n`*.

    If the source server from which the new one was cloned had already disabled hostname verification, you do not need to take this step because the hostname verification settings propagated to the cloned server.

    To disable hostname verification, set **Hostname Verification** to `None` then click **Save**.
11. Click **Activate configuration** from the Change Center menu.

Registering the New Managed Server

To configure the new managed server as an OAM server, use the Oracle Access Management Console:

1. Log in to the Oracle Access Management Console as the `oamadmin` user at `http://oamhost1.example.com:7001/oamconsole`
2. Click the **Configuration** tab. Click **Server Instances**.
3. Select **Create** from the Actions menu.
4. Enter the following information:
   - **Server Name**: `WLS_OAM3`
   - **Host**: Host that the server will run on
   - **Port**: Listen port that was assigned when the managed server was created, for example, 14100
   - **Proxy Server ID**: `AccessServerConfigProxy`
   - **Port**: Port you want the OAM proxy to run on. This is unique for the host, for example, 5575
   - **Mode**: `Open`
5. Click **Apply** when a prompt requests that you confirm the edit.
6. Ensure that the `OAM_ORACLE_HOME` property `<ORACLE_HOME>/oam` is set while starting server nodes (OAM1, OAM2 etc). In this environment, `startWeblogic` script is edited to pass `-DOAM_ORACLE_HOME=<ORACLE_HOME>/oam` while starting the Java process.

Configuring WebGate with the New OAM Managed Server

To configure the WebGate with the new OAM Managed Server, take these steps:

1. Verify that Node Manager is running on the new Access Server WLS_OAM3.
2. Start the Managed Server using the Administration Console. See the Start the Managed Server
3. Inform WebGates about the new Managed Server. See Inform WebGates of the New Managed Server

Start the Managed Server

To start the Managed Server using the Administration Console:

1. Change to the directory to OAM Domain HOME. For example, `DOMAIN_HOME/bin`
2. Start the Managed Server. For example, enter:

   `./startManagedWebLogic.sh WLS_OAM3 http://`*`hostname`*`:7001`
3. At the prompt, enter the WebLogic username and password. Click **Enter**.
4. Verify that the Managed Server is running. Check the startManagedWebLogic logs, or click **Servers** under **Environment** in the Administration Console to view the Summary page. Refresh the page to see updates.

Inform WebGates of the New Managed Server

To inform any WebGates about the new Managed Server:

1. Log in to the Oracle Access Management Console at `http://OAMHOST1.example.com:7001/oamconsole` as the `oamadmin` user.
2. Click **Application Security** tab, click **Agents** to open SSO Agents page.
3. On the SSO Agents page, click **Search**.
4. Click **Search**.
5. Click the WebGate you want to change.
6. Add the new server to either the primary or secondary server list by clicking the Add **+** icon.
7. Select the server name from the list.
8. Click **Apply**

> **Note:**
>
> Repeat this procedure to inform all the configured WebGate Agents.

Scaling Out Access Manager

You *scale out* to add a new Access Manager managed server to a new node. Scale out is very similar to scale up, but requires the software to be installed on the new node.

1. Install Oracle WebLogic Server on the new host. See Installing Oracle WebLogic Server.
2. Install Identity Management components on the new host. See Installing and Configuring the Access Manager Application Tier.
3. Log in to the Administration Console at `http://`*hostname*`.example.com:7001/oamconsole`.
4. From the Domain Structure window of the Administration Console, expand the **Environment** node and then **Machines**.
5. From the Machines table, click **New**.
6. At the Create a New Machine screen labeled Machine Identity, enter the following information:
   - **Name**: New node host name, for example, *host.example*`.com`
   - **Machine OS:** Select operating system, for example, UNIX
7. Click **Next**.
8. In the Create New Machine screen labeled **Node Manager Properties,** enter this information
   - **Type**: Keep the default SSL
   - **Listen Address**: Replace localhost with the hostname that `WLS_OAM3` will run on.
   - **Port**: Verify that the Listen Port matches the Node Manager port that will run on the other node, for example, `WLS_OAM3`.
9. Click **Finish**.
10. From the Domain Structure expand Servers.
11. Select a server on the host you want to extend, for example: WLS_OAM1.
12. Click Clone.
13. From the Clone a Server screen labeled Server Identity enter the following:
    - **Server Name**: New name for the server, for example `WLS_OAM3`.
    - **Server Listen Address**: Name of the host the Managed Server will run on.
    - **Server Listen Port**: Port the new managed server will use. This port must be unique within the host.
14. Click **OK**.
15. From the Servers table, click the new clone you just created, for example `WLS_OAM3`.
16. From the Machine option, assign the server to the new machine name you just created. This is the machine that the Managed Server will run on.
17. Click **Save**.
18. Click on the **SSL** tab.
19. Click **Advanced**.

20. Set **Hostname Verification** to **None**.
21. Click **Save**.
22. Run `pack.sh` and `unpack.sh` scripts located at `ORACLE_HOME/oracle_common/common/bin` to pack the domain on OAMHOST1 and unpack it on the new host respectively.

```
pack.sh -domain=ORACLE_HOME/user_projects/domains/domainName -
template =/tmp/idm_domain.jar -template_name="OAM Domain"
unpack.sh -domain=ORACLE_HOME/user_projects/domains/domainName -
template =/tmp/idm_domain.jar
```

- Registering the Managed Server with OAM
- Configuring WebGate with the New OAM Access Server

Registering the Managed Server with OAM

To register the new managed server as an OAM server:

1. Log in to the Oracle Access Management Console at `http://OAMHOST1.example.com:7001/oamconsole` as the `oamadmin` user.
2. Click the **Configuration** tab. Click **Server Instances**.
3. Select **Create** from the Actions menu.
4. Enter the following information:
   - **Server Name**: Enter the same server name you entered while cloning the OAM server node in the WebLogic Console.
   - **Host**: Host that the server will run on, `OAMHOST3`.
   - **Port**: Listen port that was assigned when you created the managed server.
   - **OAM Proxy Port**: Port you want the OAM proxy to run on. This is unique for the host.
   - **Proxy Server ID:** `AccessServerConfigProxy`
   - **Mode**: Select the appropriate mode: `Open`, `Simple`, or `Cert`.
5. Click **Apply**.
6. Edit `oam-config.xml` available at `<DOMAIN_HOME>/config/fmwconfig` to set the IP range of nodes that get added dynamically.

   The Settings Map for the following example is `SetWellKnownAddress>AuthorizedSubnets >Range1 >From [value of start ip range]]>To [value of end ip range].`

```
<Setting Name="AuthorizedSubnets" Type="htf:map">
<Setting Name="Range1" Type="htf:map">
<Setting Name="From" Type="htf:map">
<Setting Name="Key"
Type="xsd:string">oam.coherence.auth.range.from.1</Setting>
<Setting Name="Value"
Type="xsd:string">10.229.139.20</Setting>
</Setting>
<Setting Name="To" Type="htf:map">
```

```
<Setting Name="Key"
Type="xsd:string">oam.coherence.auth.range.to.1</Setting>
<Setting Name="Value"
Type="xsd:string">10.229.139.40</Setting>
</Setting>
</Setting>
</Setting>
```

7. Ensure that the `OAM_ORACLE_HOME` property `<ORACLE_HOME>/oam` is set while starting server nodes (OAM1, OAM2 etc). In this environment, `startWeblogic` script is edited to pass `-DOAM_ORACLE_HOME=<ORACLE_HOME>/oam` while starting the Java process.

Configuring WebGate with the New OAM Access Server

Start the Access Server. To use the server, you must inform any WebGates of its existence:

1. Log in to the Oracle Access Management Console at `http://OAMHOST1.example.com:7001/oamconsole` as the `oamadmin` user.
2. Click **Application Security** tab.
3. Click **Agents** to open SSO Agents page
4. On the SSO Agents page, click **Search**.
5. Click the WebGate you want to change.
6. Under the Server Lists section, add the new OAM Access Server `WLS_OAM3` to either the primary or secondary server list by clicking the Add [**+**] icon.
7. Select the server name from the list.
8. Click **Apply**.

Verifying the WebGate Configuration is Updated

To verify the WebGate configuration

1. Log into the Web server where the WebGate was updated previously.
2. Go to the directory `OHSDomain/config/fmwconfig/components/OHS/<instancename>/webgate/config`
3. Open `ObAccessClient.xml` with a text editor. Verify that `primary_server_list` or `secondary_server_list` shows that the new OAM Access Server is updated.

> **Note:**
>
> If the WebGate configuration does not update, recycle the web server, which pulls Webgate Agent profile updates to the `ObAccessClient.xml` file.

Editing Oracle HTTP Server Configuration File

Now that you created and started the new Managed Server, the web tier starts to direct requests to it. However, Oracle recommends informing the web server about the new Managed Server.

In the Web tier, there are several configuration files including `admin_vh.conf`, `sso_vh.conf` and `igdinternal_vh.conf` reside in the directory: *ORACLE_INSTANCE*`/config/OHS/`*component name*`/moduleconf`. Each contain a number of entries in location blocks. If a block references two server instances and you add a third one, you must update that block with the new server.

Add the new server to the `WebLogicCluster` directive in the file. For example, change:

```
<Location /oam>
   SetHandler weblogic-handler
   WebLogicCluster OAMHOST1.example.com:14100,OAMHOST2.example.com:14100
</Location>
```

to:

```
<Location /oam>
   SetHandler weblogic-handler
   WebLogicCluster
OAMHOST1.example.com:14100,OAMHOST2.example.com:14100,OAMHOST1.example.com:14101
</Location>
```

Deploying Oracle Identity and Access Management cluster with Unicast configuration

If multicast IP is disabled in deployment environment then you can deploy Oracle Identity and Access Management cluster with Unicast configuration.

In your deployment environment, if multicast IP is disabled because of security reasons or if you are using cloud Infrastructure for deployment, it is not feasible to deploy Oracle Identity and Access Management with the default configuration (multicast). It is not feasible because Oracle Identity and Access Management 12c uses EH cache, which depends on JavaGroup or JGroup library and supports multicast configuration as default for messages broadcasting.

To configure unicast configuration for EH cache, complete the following steps:

1. Get the list of host machines and available port to prepare the below JGroup configuration.

   Example:

   ```
   TCP(bind_port=7800;loopback=true):TCPPING(timeout=3000;initial_hosts=1
   .2.3.4[7800],1.2.3.5[7800];port_range=5;num_initial_members=2):pbcast.
   NAKACK(use_mcast_xmit=false;gc_lag=20;
   retransmit_timeout=1000):pbcast.GMS(print_local_addr=true;join_timeout
   =3000)
   ```

   In the example, you have servers: `1.2.3.4` and `1.2.3.5` and the available port on these machines is `7800`.

2. In the EM Console, expand **Identity and Access** > **OIM**.
3. Right-click on **oim(*version_number*)** and select **System MBean Browser**.

   Where, *version_number* is the current version number of Oracle Identity and Access Management.

4. Expand **oracle.iam** > **Server:oim** > **Application:oim** > **XMLConfig** > **Config** > **XMLConfig.CacheConfig** > **Cache**.

5. In the right pane, **Attributes** tab, set the **Clustered** attribute value to `true`.
6. In the left pane, expand the **Cache** folder and select **XMLConfig.CacheConfig.XLCacheProvider** > **XLCacheProvider**.
7. In the right pane, **Attributes** tab, set the **MulicastConfig** attribute value to the equivalent JGroup string you identified in step 1.
8. Click **Apply**.
9. Restart all the Oracle Identity and Access Management managed servers.

8

Uninstalling or Reinstalling Oracle Identity and Access Management

Follow the instructions in this section to uninstall or reinstall Oracle Identity and Access Management.

Oracle recommends that you always use the instructions in this section to remove the software. If you try to remove the software manually, you may encounter problems when you try to reinstall the software again at a later time. Following the procedures in this section ensures that the software is properly removed.

- About Product Uninstallation
  The Oracle Fusion Middleware uninstaller removes the software from the Oracle home directory.
- Stopping Oracle Fusion Middleware
  Before running the Uninstall Wizard, Oracle recommends that you stop all servers and processes associated with the Oracle home you are going to remove.
- Removing Your Database Schemas
  Before you remove the Oracle home, Oracle recommends that you run the Repository Creation Utility (RCU) to remove database schemas associated with this domain.
- Uninstalling the Software
  Follow the instructions in this section to start the Uninstall Wizard and remove the software.
- Removing the Oracle Home Directory Manually
  After you uninstall the software, you must manually remove your Oracle home directory and any existing subdirectories that the Uninstall Wizard did not remove.
- Removing the Program Shortcuts on Windows Operating Systems
  On Windows operating systems, you must also manually remove the program shortcuts; the Deinstallation Wizard does not remove them for you.
- Removing the Domain and Application Data
  After you uninstall the software, you must remove the domain and application data.
- Reinstalling the Software
  You can reinstall your software into the same Oracle home as a previous installation only if you uninstalled the software by following the instructions in this section, including manually removing the Oracle home directory.

About Product Uninstallation

The Oracle Fusion Middleware uninstaller removes the software from the Oracle home directory.

The following table summarizes the tasks to uninstall Fusion Middleware products.

Table 8-1 Roadmap for Product Uninstallation

Task	Description	Documentation
Stop Oracle Fusion Middleware	All servers and processes in your domain should be stopped before running the uninstaller.	See Stopping Oracle Fusion Middleware.
Remove your database schemas	Run Repository Creation Utility to remove your database schemas.	See Removing Your Database Schemas.
Remove the software	Run the product uninstaller to remove Oracle Fusion Middleware Infrastructure. Note that if your Oracle home contains multiple products, you must run the uninstaller multiple times, once for each product.	See Uninstalling the Software.
Remove the Oracle home directory	The uninstaller does not remove all files and folders from the Oracle home directory. After the uninstaller is finished, you must manually remove the Oracle home to complete your product removal.	See Removing the Oracle Home Directory Manually.
Remove your domain and application data	The uninstaller does not remove data contained in your Domain home or Application home directories, even if they are located inside the Oracle home. You must remove these directories manually.	See Removing the Domain and Application Data.

Stopping Oracle Fusion Middleware

Before running the Uninstall Wizard, Oracle recommends that you stop all servers and processes associated with the Oracle home you are going to remove.

See Stopping an Oracle Fusion Middleware Environment in *Administering Oracle Fusion Middleware*.

Removing Your Database Schemas

Before you remove the Oracle home, Oracle recommends that you run the Repository Creation Utility (RCU) to remove database schemas associated with this domain.

Each domain has its own set of schemas, uniquely identified by a custom prefix. For more information about custom prefixes, see About Custom Prefixes in *Creating Schemas with the Repository Creation Utility*. This set of schemas cannot be shared with any other domain. For more information about creating schemas with the RCU, see Planning Your Schema Creation in *Creating Schemas with the Repository Creation Utility*.

If there are multiple sets of schemas on your database, be sure to identify the schema prefix associated with the domain that you are removing.

For schema removal steps, see Dropping Schemas in *Creating Schemas with the Repository Creation Utility*.

Uninstalling the Software

Follow the instructions in this section to start the Uninstall Wizard and remove the software.

If you want to uninstall the product in a silent (command-line) mode, see Running the Oracle Universal Installer for Silent Uninstallation in *Installing Software with the Oracle Universal Installer*.

Starting the Uninstall Wizard

To start the Uninstall Wizard:

1. Change to the following directory:

   (UNIX) *ORACLE_HOME*`/oui/bin`

   (Windows) *ORACLE_HOME*`\oui\bin`

2. Enter the following command:

   (UNIX) `./deinstall.sh`

   (Windows) `deinstall.cmd`

Selecting the Product to Uninstall

Because multiple products exist in the Oracle home, ensure that you are uninstalling the correct product.

After you run the Uninstall Wizard, the Distribution to Uninstall screen opens. From the dropdown menu, select `Oracle Identity and Access Management` and click **Uninstall**. The uninstallation program shows the screens listed in Navigating the Uninstall Wizard Screens.

> **Note:**
>
> You can uninstall Oracle Fusion Middleware Infrastructure after you uninstall Oracle Identity and Access Management software by running the Uninstall Wizard again. Before doing so, make sure that there are no other products using the Infrastructure; those products will no longer function once the Infrastructure is removed. You will not encounter the Distribution to Uninstall screen if no other software depends on Oracle Fusion Middleware Infrastructure. See Uninstalling Oracle Fusion Middleware Infrastructure in *Installing and Configuring the Oracle Fusion Middleware Infrastructure*.

Navigating the Uninstall Wizard Screens

The Uninstall Wizard shows a series of screens to confirm the removal of the software.

If you need help on screen listed in Table 8-2, click **Help** on the screen.

Table 8-2 Uninstall Wizard Screens and Descriptions

Screen	Description
Welcome	Introduces you to the product Uninstall Wizard.
Uninstall Summary	Shows the Oracle home directory and its contents that are uninstalled. Verify that this is the correct directory. If you want to save these options to a response file, click **Save Response File** and enter the response file location and name. You can use the response file later to uninstall the product in silent (command-line) mode. See Running the Oracle Universal Installer for Silent Uninstall in *Installing Software with the Oracle Universal Installer*. Click **Deinstall**, to begin removing the software.
Uninstall Progress	Shows the uninstallation progress.
Uninstall Complete	Appears when the uninstallation is complete. Review the information on this screen, then click **Finish** to close the Uninstall Wizard.

Removing the Oracle Home Directory Manually

After you uninstall the software, you must manually remove your Oracle home directory and any existing subdirectories that the Uninstall Wizard did not remove.

For example, if your Oracle home directory is `/home/Oracle/product/ORACLE_HOME` on a UNIX operating system, enter the following commands:

```
cd /home/Oracle/product
rm -rf ORACLE_HOME
```

On a Windows operating system, if your Oracle home directory is `C:\Oracle\Product\ORACLE_HOME`, use a file manager window and navigate to the `C:\Oracle\Product` directory. Right-click on the *ORACLE_HOME* folder and select **Delete**.

Removing the Program Shortcuts on Windows Operating Systems

On Windows operating systems, you must also manually remove the program shortcuts; the Deinstallation Wizard does not remove them for you.

To remove the program shortcuts on Windows:

1. Change to the following directory:
   `C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Oracle\ORACLE_HOME\Product`
2. If you only have one product installed in your Oracle home, delete the *ORACLE_HOME* directory. If you have multiple products installed in your Oracle home, delete all products before you delete the *ORACLE_HOME* directory.

Removing the Domain and Application Data

After you uninstall the software, you must remove the domain and application data.

To remove the domain and application data:

1. Manually remove your Domain home directory. For example:

   On a UNIX operating system, if your Domain home directory is `/home/Oracle/config/domains/idm_domain`, enter the following command:

   ```
   cd /home/Oracle/config/domains
   ```

   ```
   rm -rf idm_domain
   ```

   On a Windows operating system, if your Domain home directory is `C:\Oracle\Config\domains\idm_domain`, use a file manager window and navigate to the `C:\Oracle\Config\domains` directory. Right-click on the `idm_domain` folder and select **Delete**.

2. Manually remove your Application home directory. For example:

   On a UNIX operating system, if your Application home directory is `/home/Oracle/config/applications/idm_domain`, enter the following commands:

   ```
   cd /home/Oracle/config/applications
   ```

   ```
   rm -rf idm_domain
   ```

   On a Windows operating system, if your Application home directory is `C:\Oracle\Config\applications\idm_domain`, use a file manager window and navigate to the `C:\Oracle\Config\applications` directory. Right-click on the `idm_domain` folder and select **Delete**.

3. Back up the `domain_registry.xml` file in your Oracle home, then edit the file and remove the line associated with the domain that you are removing. For example, to remove the `idm_domain`, find the following line and remove it:

   ```
   <domain location="/home/Oracle/config/domains/idm_domain"/>
   ```

   Save and exit the file when you are finished.

Reinstalling the Software

You can reinstall your software into the same Oracle home as a previous installation only if you uninstalled the software by following the instructions in this section, including manually removing the Oracle home directory.

When you reinstall, you can then specify the same Oracle home as your previous installation.

Consider the following cases where the Oracle home is not empty:

- Installing in an existing Oracle home that contains the same feature sets.

  The installer warns you that the Oracle home that you specified during installation already contains the same software you are trying to install.

- Installing in an existing, non-empty Oracle home.

  For example, suppose you chose to create your Domain home or Application home somewhere inside your existing Oracle home. This data is not removed when you

uninstall a product, so if you try to reinstall into the same Oracle home, the installer does not allow it. Your options are:

- Uninstall your software from the Oracle home (as this section describes) and then remove the Oracle home directory. After you uninstall the software and remove the Oracle home directory, you can reinstall and reuse the same Oracle home location. Any domain or application data that was in the Oracle home must be re-created.
- Select a different Oracle home directory.

A

Updating the JDK After Installing and Configuring an Oracle Fusion Middleware Product

Consider that you have a JDK version jdk1.8.0_191 installed on your machine. When you install and configure an Oracle Fusion Middleware product, the utilities, such as Configuration Wizard (`config.sh|exe`), OPatch, or RCU point to a default JDK, for example, jdk1.8.0_191. After some time, Oracle releases a new version of the JDK, say jdk1.8.0_211 that carries security enhancements and bug fixes. You can upgrade the existing JDK to a newer version, and can have the complete product stack point to the newer version of the JDK.

You can maintain multiple versions of JDK and switch to the required version on need basis.

- About Updating the JDK Location After Installing an Oracle Fusion Middleware Product
  The binaries and other metadata and utility scripts in the Oracle home and Domain home, such as RCU or Configuration Wizard, use a JDK version that was used while installing the software and continue to refer to the same version of the JDK. The JDK path is stored in a variable called JAVA_HOME which is centrally located in .globalEnv.properties file inside the *`ORACLE_HOME`*`/oui` directory.

About Updating the JDK Location After Installing an Oracle Fusion Middleware Product

The binaries and other metadata and utility scripts in the Oracle home and Domain home, such as RCU or Configuration Wizard, use a JDK version that was used while installing the software and continue to refer to the same version of the JDK. The JDK path is stored in a variable called JAVA_HOME which is centrally located in .globalEnv.properties file inside the *`ORACLE_HOME`*`/oui` directory.

The utility scripts such as config.sh|cmd, launch.sh, or opatch reside in the *ORACLE_HOME*, and when you invoke them, they refer to the JAVA_HOME variable located in .globalEnv.properties file. To point these scripts and utilities to the newer version of JDK, you must update the value of the JAVA_HOME variable in the .globalEnv.properties file by following the directions listed in Updating the JDK Location in an Existing Oracle Home .

To make the scripts and files in your Domain home directory point to the newer version of the JDK, you can follow one of the following approaches:

- Specify the path to the newer JDK on the Domain Mode and JDK screen while running the Configuration Wizard.

  For example, consider that you installed Oracle Fusion Middleware Infrastructure with the JDK version 8u191. So while configuring the WebLogic domain with the Configuration Assistant, you can select the path to the newer JDK on the Domain Mode and JDK screen of the Configuration Wizard. Example: `/scratch/jdk/jdk1.8.0_211`.

- Manually locate the files that have references to the JDK using `grep` (UNIX) or `findstr` (Windows) commands and update each reference. See Updating the JDK Location in an Existing Oracle Home .

> **Note:**
>
> If you install the newer version of the JDK in the same location as the existing JDK by overwriting the files, then you don't need to take any action.

When you upgrade Oracle Identity Manager in an integrated environment, you may encounter the OPSS processing error. The following exception is seen when you run reconfig.sh command to reconfigure the Oracle Identity Manager domain:

```
SEVERE [93] com.oracle.cie.domain.progress.AbstractProgressGenerator -
Error occurred in
phase {OPSS Processing} execution.
java.lang.IllegalStateException: SecurityContext: Domain Name:
IAMGovernanceDomain
JDBC URL: opss-audit-DBDS:jdbc:oracle:thin:@//slc03rmj:1521/IDMDB
JDBC URL: opss-data-source:jdbc:oracle:thin:@//slc03rmj:1521/idmdb
le.com
Caused by: java.security.InvalidKeyException: Illegal key size
at javax.crypto.Cipher.checkCryptoPerm(Cipher.java:1039)
at javax.crypto.Cipher.implInit(Cipher.java:805)
at javax.crypto.Cipher.chooseProvider(Cipher.java:864)
at javax.crypto.Cipher.init(Cipher.java:1396)
at javax.crypto.Cipher.init(Cipher.java:1327)
```

To resolve this issue:

1. Install the Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files from the following location: Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 8 Download.
2. Copy the local_policy.jar and the US_export_policy.jar files to the location `JAVA_HOME/jre/lib/security/`. If the files already exist in the destination folder, overwrite them.

- Updating the JDK Location in an Existing Oracle Home
  The `getProperty.sh|cmd` script displays the value of a variable, such as JAVA_HOME, from the .globalEnv.properties file. The `setProperty.sh|cmd` script is used to set the value of variables, such as OLD_JAVA_HOME or JAVA_HOME that contain the locations of old and new JDKs in the .globalEnv.properties file.
- Updating the JDK Location in an Existing Domain Home
  You must search the references to the current JDK, for example 1.8.0_191 manually, and replace those instances with the location of the new JDK.

Updating the JDK Location in an Existing Oracle Home

The `getProperty.sh|cmd` script displays the value of a variable, such as JAVA_HOME, from the .globalEnv.properties file. The `setProperty.sh|cmd` script is used to set the

value of variables, such as OLD_JAVA_HOME or JAVA_HOME that contain the locations of old and new JDKs in the .globalEnv.properties file.

The `getProperty.sh|cmd` and `setProperty.sh|cmd` scripts are located in the following location:

(UNIX) *ORACLE_HOME*`/oui/bin`
(Windows) *ORACLE_HOME*`\oui\bin`
Where, *ORACLE_HOME* is the directory that contains the products using the current version of the JDK, such as 1.8.0_191.

To update the JDK location in the .globalEnv.properties file:

1. Use the `getProperty.sh|cmd` script to display the path of the current JDK from the JAVA_HOME variable. For example:

   (UNIX) *ORACLE_HOME*`/oui/bin/getProperty.sh JAVA_HOME`
   (Windows) *ORACLE_HOME*`\oui\bin\getProperty.cmd JAVA_HOME`
   `echo JAVA_HOME`
   Where JAVA_HOME is the variable in the .globalEnv.properties file that contains the location of the JDK.

2. Back up the path of the current JDK to another variable such as OLD_JAVA_HOME in the .globalEnv.properties file by entering the following commands:

   (UNIX) *ORACLE_HOME*`/oui/bin/setProperty.sh -name OLD_JAVA_HOME -value` *specify_the_path_of_current_JDK*
   (Windows) *ORACLE_HOME*`\oui\bin\setProperty.cmd -name OLD_JAVA_HOME -value` *specify_the_path_of_current_JDK*

   This command creates a new variable called OLD_JAVA_HOME in the .globalEnv.properties file, with a value that you have specified.

3. Set the new location of the JDK in the JAVA_HOME variable of the .globalEnv.properties file, by entering the following commands:

   (UNIX) *ORACLE_HOME*`/oui/bin/setProperty.sh -name JAVA_HOME -value` *specify_the_location_of_new_JDK*
   (Windows) *ORACLE_HOME*`\oui\bin\setProperty.cmd -name JAVA_HOME -value` *specify_the_location_of_new_JDK*

   After you run this command, the JAVA_HOME variable in the .globalEnv.properties file now contains the path to the new JDK, such as jdk1.8.0_211.

Updating the JDK Location in an Existing Domain Home

You must search the references to the current JDK, for example 1.8.0_191 manually, and replace those instances with the location of the new JDK.

You can use the `grep` (UNIX) or `findstr` (Windows) commands to search for the jdk-related references.

You'll likely be required to update the location of JDK in the following three files:

(UNIX) *DOMAIN_HOME*`/bin/setNMJavaHome.sh`
(Windows) *DOMAIN_HOME*`\bin\setNMJavaHome.cmd`

(UNIX) *DOMAIN_HOME*`/nodemanager/nodemanager.properties`

(Windows) *DOMAIN_HOME*\nodemanager\nodemanager.properties

(UNIX) *DOMAIN_HOME*/bin/setDomainEnv.sh
(Windows) *DOMAIN_HOME*\bin\setDomainEnv.cmd