Oracle® Fusion Middleware

Installing and Configuring Oracle Internet Directory

14c (14.1.2.1.0)
F85507-02
April 2025

ORACLE

Oracle Fusion Middleware Installing and Configuring Oracle Internet Directory, 14c (14.1.2.1.0)

F85507-02

Primary Author: Oracle Corporation

Contents

8 Uninstalling or Reinstalling Oracle Internet Directory

A Updating the JDK After Installing and Configuring an Oracle Fusion Middleware Product

Preface

This document describes how to install and configure Oracle Internet Directory.

Audience

This guide is intended for system administrators or application developers who are installing and configuring Oracle Internet Directory. 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 https://www.oracle.com/corporate/accessibility/.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit https://support.oracle.com/portal/ or visit Oracle Accessibility Learning and Support 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

Refer to the Oracle Fusion Middleware Library for additional information.

- For administering Oracle Internet Directory, see *Administering Oracle Internet Directory*.
- 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

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 Internet Directory Installation

The standard installation for Oracle Internet Directory described in this guide creates the standard topology, which represents a sample starting topology for this product.

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 Internet Directory. 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 Internet Directory 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 Internet Directory.

See About Elements in the Standard Installation Topology Illustration for information on elements of this topology.

Figure 1-1 Standard Topology for Oracle Internet Directory Standalone Installation

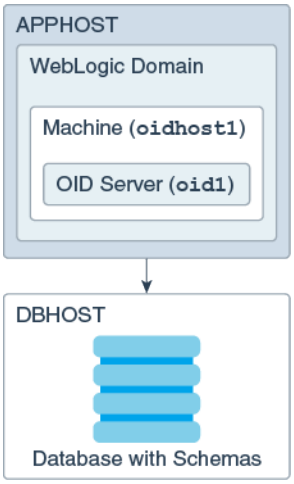
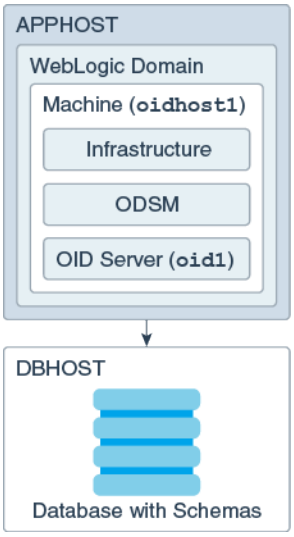


Figure 1-2 Standard Topology for Oracle Internet Directory Collocated Installation



For Oracle Internet Directory configuration instructions, see Configuring Oracle Internet Directory Domain.

For Oracle Directory Integration Platform configuration instructions, see Configuring Oracle Directory Integration Platform.

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 1-1 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*.
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*.

Table 1-1 (Cont.) Description of Elements in Standard Installation Topologies

Element	Description and Links to Related Documentation
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).

Using This Document to Extend an Existing Domain

The procedures in this guide describe how to create a new domain. The assumption is that no other Oracle Fusion Middleware products are installed on your system.

If you have installed and configured other Oracle Fusion Middleware products on your system (for example, Fusion Middleware Infrastructure, with a domain that is up and running) and wish to extend the same domain to include Oracle Internet Directory, see Installing Multiple Products in the Same Domain.

2

Preparing to Install and Configure Oracle Internet Directory

To prepare for your Oracle Internet Directory installation, verify that your system meets the basic requirements, then obtain the correct installation software.

Roadmap for Installing and Configuring a Standard Installation Topology

This roadmap provides the steps required to install and configure a standard Oracle Internet Directory installation topology.

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*.
Obtain the appropriate distributions.	Oracle Internet Directory (OID) can be installed in two modes — Standalone and Collocated. If you choose to install in standalone mode, you do not require Oracle Fusion Middleware Infrastructure. If you wish to install OID in a collocated mode, you must install Oracle Fusion Middleware Infrastructure, and ensure that the Oracle Internet Directory is installed in the same Oracle Home as Infrastructure.	See About Product Distributions.
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*.

Table 2-1 (Cont.) Standard Installation Roadmap

Task	Description	Documentation
Install prerequisite software.	If you are installing OID in a collocated mode, you must install Oracle Fusion Middleware Infrastructure 14.1.2.1.0 to create the Oracle home directory for Oracle Internet Directory. For OID standalone installation, you do not require Oracle Fusion Middleware Infrastructure.	See Installing the Infrastructure Software in *Installing and Configuring the Oracle Fusion Middleware Infrastructure*.
Install the software.	Run the Oracle Universal Installer to install Oracle Internet Directory. Installing the software transfers the software to your system and creates the Oracle home directory.	See Installing the Oracle Internet Directory Software.
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 Internet Directory 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.	See Creating the Database Schemas.
Create a WebLogic domain.	Use the Configuration Wizard/ Assistant to create and configure the WebLogic domain. This step is optional for a standalone Oracle Internet Directory installation.	See Configuring the 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 Internet Directory.

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.

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

Task	Description	Documentation
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.
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 Internet Directory.	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.

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

   Ensure that you install your product on a supported hardware and software configuration.

   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.

   > **Note:**
   >
   > This installation requires a minimum of 32GB of memory and 2 core CPU machine.

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. 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.

- 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.

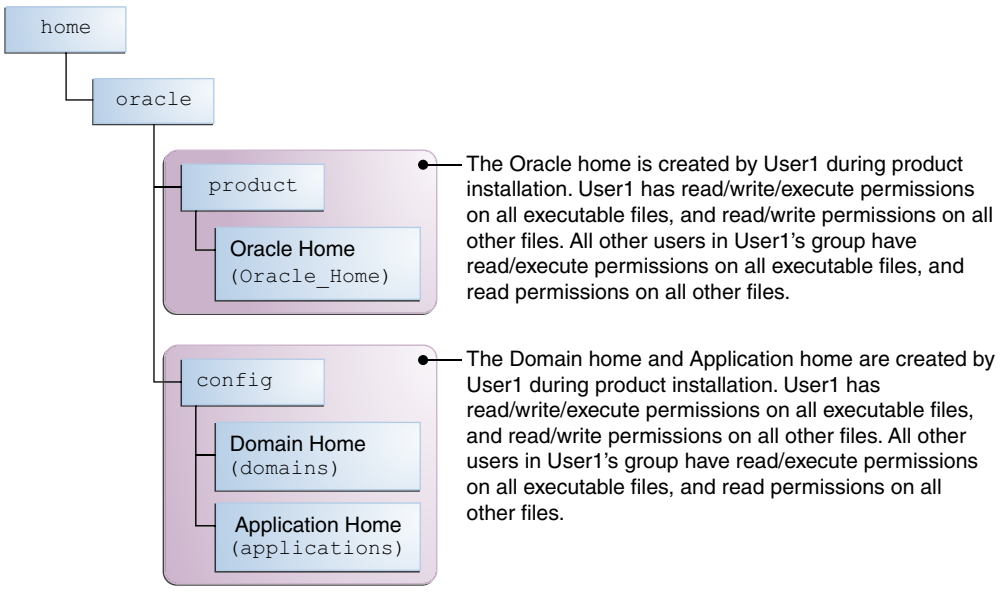
Consider the following examples:

- **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

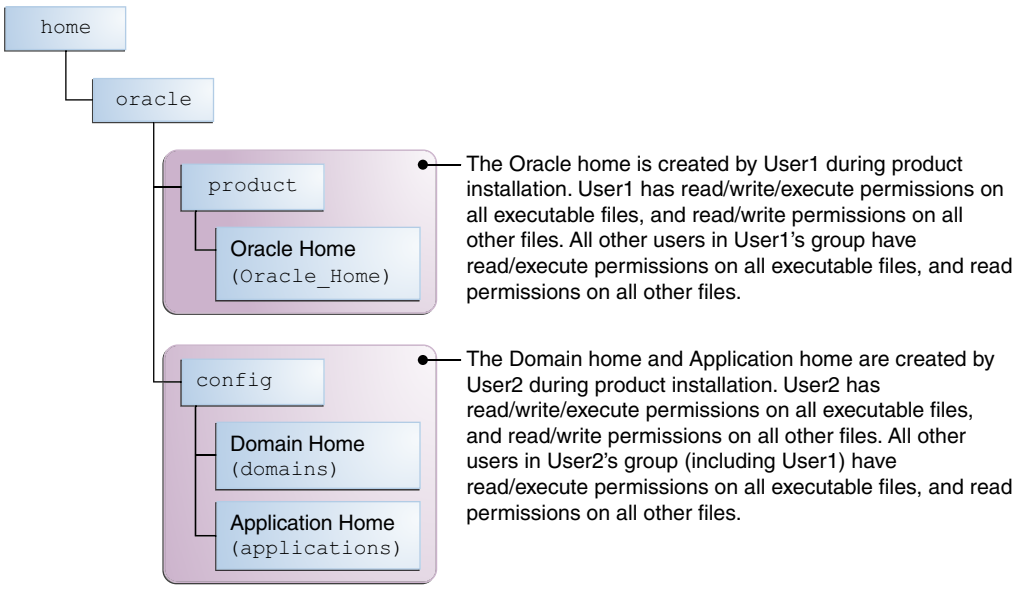


If the user who creates the domain is different than the user who installed the software, then both users must have the same privileges, as shown in the next example.

- **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 Internet Directory, ensure that you have enabled User Account Control (UAC). If you have not done so already, enable UAC by following the instructions in the *Enabling User Account Control (UAC)* section from the appropriate version of *Oracle Fusion Middleware System Requirements and Specifications* for your installation.

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

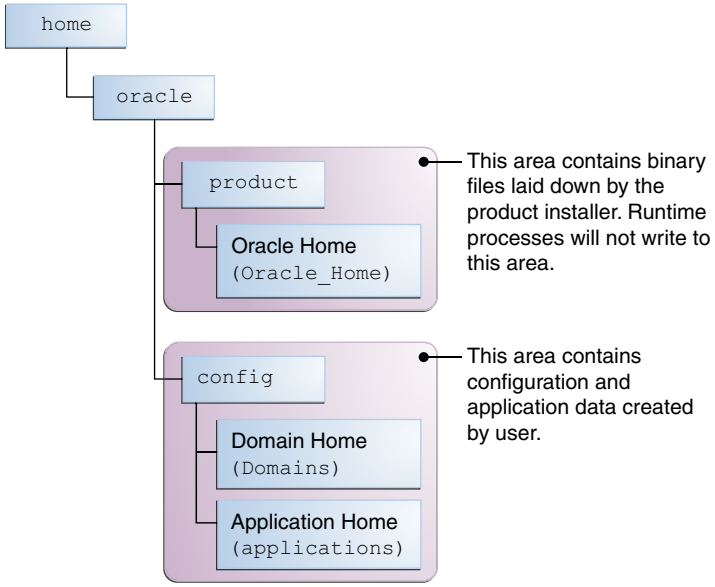
During the installation and domain configuration process, you must plan on providing the locations for these directories: Oracle home, Domain home, and the Application home.

About the Recommended Directory Structure

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

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, you 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 14c (14.1.2.1.0) 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.

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`.

Installing Multiple Products in the Same Domain

There are two methods to install and configure multiple products in one domain. This is also known as *extending* a domain.

- **Method 1.**

  Install and configure Product A, including creating the schemas and starting all servers in the domain to verify a successful domain configuration.

  This is the method used in all installation guides in the Fusion Middleware library. You can repeat this process for as many products as necessary. It allows you to validate one product at a time and add more products incrementally.

  To install Product B in the same domain as Product A:

  1. Stop all servers to prevent any updates to the domain while you add the new product.

     See Starting and Stopping Oracle Fusion Middleware in *Administering Oracle Fusion Middleware*.

  2. Follow the instructions in the installation guide for Product B, including creating the necessary schemas.

  3. Run the Configuration Wizard to configure the domain.

     During configuration, the Configuration Wizard automatically detects the components that have been installed and offers you the option to extend the existing Product A domain to include Product B.

- **Method 2.**

Install all of the required products, then create the schemas for all of the products. After you create the schemas, configure the domain by using the necessary product templates, then start all the servers.

This method of creating a multi-product domain may be slightly faster than Method 1; however, the installation guides in the Fusion Middleware library do not provide specific instructions for this method of domain creation.

See Also:

- To update WebLogic domains, see Updating WebLogic Domains in *Creating WebLogic Domains Using the Configuration Wizard*.
- For important information regarding the ability of Oracle Fusion Middleware products to function with previous versions of other Oracle Fusion Middleware, Oracle, or third-party products, see Oracle Fusion Middleware 14c (14.1.2.1.0) Interoperability and Compatibility in *Understanding Interoperability and Compatibility*.

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 14c (14.1.2.1.0), the certified JDK is 17.0.12 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 the Oracle Technology Network (OTN).

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.

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*.

About Product Distributions

You create the initial Oracle Internet Directory domain using the Oracle Fusion Middleware Infrastructure distribution, which contains both Oracle WebLogic Server software and Oracle Java Required Files (JRF) software.

Oracle JRF software consists of:

- Oracle Web Services Manager
- Oracle Application Development Framework (Oracle ADF)
- Oracle Enterprise Manager Fusion Middleware Control
- Repository Creation Utility (RCU)
- Other libraries and technologies required to support Oracle Fusion Middleware products

Prerequisites:

- Install Oracle Fusion Middleware Infrastructure. For more information about installing Oracle Fusion Middleware Infrastructure, see Installing the Infrastructure Software in *Installing and Configuring the Oracle Fusion Middleware Infrastructure*.
- For SUSE 11 or later:
  - The `openmotif` package is not included by default on SUSE 11 or later. You need the `openmotif` package installed to successfully install Oracle Internet Directory on SUSE 11 or later.

    Obtain this package from the Novell website and then perform the installation using the instructions provided by Novell.
  - Create a soft-link from `/lib64/libnsl.so` to `/lib64/libnsl.so.1`.

Obtaining the Product Distribution

You can obtain the Oracle Fusion Middleware Infrastructure and Oracle Internet Directory distribution on Oracle Technology Network or Oracle Software Delivery Cloud.

To prepare to install Oracle Fusion Middleware Infrastructure and Oracle Internet Directory:

1. Enter `java -version` on the command line to verify that a certified JDK is installed on your system. For 14c (14.1.2.1.0), the certified JDK is 17.0.12 and later.

   See About JDK Requirements for an Oracle Fusion Middleware Installation.

2. Locate and download the Oracle Fusion Middleware Infrastructure and Oracle Internet Directory software.

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

   To obtain the distribution for product evaluation, visit the Oracle Software Delivery Cloud page.

After preparing to install and configure the software, see Installing the Oracle Internet Directory Software.

3

Installing the Oracle Internet Directory Software

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

> **Note:**
>
> The latest Oracle Identity Management 14.1.2.1.0 Stack Patch Bundle should be applied. For more information, see the Stack Patch Bundle for Oracle Identity Management Products (Doc ID 2657920.2) at https://support.oracle.com.

If you wish to install Oracle Internet Directory in `Standalone` mode, you do not require Oracle Fusion Middleware Infrastructure. You can proceed with the Oracle Internet Directory installation.

If you wish to install Oracle Internet Directory in `Collocated` mode, ensure that you install Oracle Fusion Middleware Infrastructure 14*c* (14.1.2.0.0) first, followed by the Oracle Internet directory 14c (14.1.2.1.0). Infrastructure and Oracle Internet Directory must be installed in the same Oracle Home.

For more information about installing Oracle Fusion Middleware Infrastructure 14*c* (14.1.2.0.0), see Installing the Infrastructure Software in the *Installing and Configuring the Oracle Fusion Middleware Infrastructure*.

Verifying the Installation Checklist

The installation process requires specific information.

Table 3-1 lists important items that you must know before, or decide during, Oracle Internet Directory installation.

Table 3-1 Installation Checklist

Information	Example Value	Description
`JAVA_HOME`	`/u01/oracle/jdk17.0.12`	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`.

Table 3-1 (Cont.) Installation Checklist

Information	Example Value	Description
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	`password`	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 Internet Directory, as needed.
WebLogic Server hostname	`examplehost.exampledomain`	Host name for Oracle WebLogic Server and Oracle Internet Directory consoles.

Table 3-1 (Cont.) Installation Checklist

Information	Example Value	Description
Console port	**Note:** The default port values will vary depening on how you conifigured your domain. For a list of default values, see Port Numbers by Product and Component.	Port for Oracle Internet Directory consoles.
DOMAIN_HOME	`/home/Oracle/config/domains/oid_domain`	Location in which your domain data is stored.
APPLICATION_HOME	`/home/Oracle/config/applications/oid_domain`	Location in which your application data is stored.

Table 3-1 (Cont.) Installation 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	`password`	Password of the user with Oracle WebLogic Server administration privileges.
RCU	*`ORACLE_HOME`*`/oracle_common/bin`	Path to the Repository Creation Utility (RCU).

Table 3-1 (Cont.) Installation Checklist

Information	Example Value	Description
RCU schema prefix	`oid`	Prefix for names of database schemas used by Oracle Internet Directory.

Note:

The schema prefix is not required for the Oracle Internet Directory schema (ODS) irrespective of the installation type (standalone or collocated). Prefix is only required for the

Table 3-1 (Cont.) Installation Checklist

Information	Example Value	Description
		other schemas that are created along with ODS schema.
RCU schema password	`password`	Password for the database schemas used by Oracle Internet Directory.
Configuration utility	*ORACLE_HOME*`/oracle_common/common/bin`	Path to the Configuration Wizard for domain creation and configuration.

Starting the Installation Program

You can start the installation program on UNIX or Windows.

To start the installation program:

1. Sign in to the host system.
2. Go to the directory where you have extracted the contents of product distribution archive file.
3. Enter the following command:
   - (UNIX) `./fmw_14.1.2.1.0_oid_linux64.bin`
   - (Windows) `setup_fmw_14.1.2.1.0_oid_win64.exe`

> **Note:**
>
> You will not be able to execute `./fmw_14.1.2.1.0_oid_linux64.bin` if it does not have execute permission. Make sure to check and grant execute permission before running this command.

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

> **WARNING:**
>
> Ignore any warnings stating that installing OID 14.1.2.1.0 on AIX 7.3 is not certified.

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*.
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.
Installation Type	Select **Standalone OID** or **Collocated OID** based on what topology you would like to deploy. In case of a Standalone mode, you can install OID without configuring any WebLogic domain. If you choose Collocated mode, OID will be managed by WebLogic domain. You will have to install Oracle Fusion Middleware Infrastructure 14c (14.1.2.0.0) prior to installing OID, in case of a Collocated mode.
JDK Selection	**Note**: This screen appears for certain distributions only. Use this screen to select the JDK to use for this installation.
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.

Table 3-2 (Cont.) Install Screens

Screen	Description
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.

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*.

4

Configuring Oracle Internet Directory Domain

After you have installed Oracle Internet Directory, 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 Oracle Internet Directory
- Installing the Oracle Internet Directory 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.

> **Note:**
>
> As of Oracle Fusion Middleware 14c (14.1.2.1.0), new schemas are created with editions-based redefinition (EBR) views enabled by default. Oracle Internet Directory schemas do not support EBR, therefore, in order to use the EBR functionality with your non-OAM schemas, you will have to run the RCU twice.
>
> When EBR is enabled, the schema objects can be upgraded online to a future Fusion Middleware release without any downtime. For more information about using editions-based redefinition, see Using Edition-based Redefinition.

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 Autonomous Transaction Processing-Dedicated (ATP-D) and Autonomous Transaction Processing-Dedicated (ATP-D)), you must modify the wallet settings and set the environment variables, and apply patches on `ORACLE HOME`. For more information, see Settings to connect to Autonomous Transaction Processing Database for Oracle Internet Directory 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 14c (14.1.2.1.0), the certified JDK is 17.0.12 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.
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.

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, select **System Load and Product Load**. This procedure assumes that you have the necessary permissions and privileges to perform DBA activities on your database, that is the SYSDBA privileges.

> **Note:**
>
> For an Autonomous Transaction Processing database (both Autonomous Transaction Processing-Dedicated (ATP-D) and Autonomous Transaction Processing-Dedicated (ATP-D)), 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.

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 Autonomous Transaction Processing-Dedicated (ATP-D) database, you must use only one of the database service names, `<databasename>_tpurgent` or `<databasename>_tp`, specified in `tnsnames.ora`.

To create schemas on an Autonomous Transaction Processing database (both Autonomous Transaction Processing-Dedicated (ATP-D) and Autonomous Transaction Processing-Dedicated (ATP-D)), 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.

> **Note:**
>
> You must invoke RCU twice. When you invoke RCU the first time select **Database type** as **Oracle Database enabled for edition-based redefinition** and load the EBR dependent components (STB, OPSS, IAU, IAU_Append, IAU_Viewer, and WLS). When you invoke RCU the second time, select **Database type** as **Oracle Database** and load OID.

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 Autonomous Transaction Processing-Dedicated (ATP-D) and Autonomous Transaction Processing-Dedicated (ATP-D)), use the connect string specified in `tnsnames.ora` that is present in `/<$ORACLE_HOME>/network/admin`, which is the location of the wallet files, for your service name or `TNS_alias`.

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

```
(DESCRIPTION=(CONNECT_TIMEOUT=120)(RETRY_COUNT=20)(RETRY_DELAY=3)
(TRANSPORT_CONNECT_TIMEOUT=3)(ADDRESS_LIST=(LOAD_BALANCE=on)
(ADDRESS=(PROTOCOL=<protocol_name>)(HOST=<host_name>)(PORT=<port_number>)))
(CONNECT_DATA=(SERVICE_NAME=<service_name>.atp.oraclecloud.com)))
```

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

```
(DESCRIPTION=(CONNECT_TIMEOUT=120)=(RETRY_COUNT=20)(RETRY_DELAY=3)
(ADDRESS=(PROTOCOL=<protocol_name>)
(PORT=<port_number>)(HOST=<host_name>))
(CONNECT_DATA=(SERVICE_NAME=<service_name>.adb.oraclecloud.com))
(security=(ssl_server_cert_dn="CN=example.com,
OU=<organizational_unit>, O=<organization>, L=<city>, ST=<state>,
C=<country>")))
```

> **Note:**
>
> In this example for Autonomous Transaction Processing-Dedicated (ATP-D), you must use only one of the database service names, `<databasename>_tpurgent` or `<databasename>_tp`, specified in `tnsnames.ora`.

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 select the **Oracle Internet Directory** schema. This action automatically selects the following schemas as dependencies:

> **Note:**
>
> Oracle Internet Directory (ODS) schema does not need a prefix. The prefix is required for the other schemas selected during the schema creation process.
>
> You can load only one **Oracle Internet Directory** (ODS) schema per Database.

If you are configuring Oracle Internet Directory in a standalone mode, the following dependant schema is selected:

- Common Infrastructure Service (STB)

If you are configuring Oracle Internet Directory in a collocated mode, the following dependant schemas are selected:

- Oracle Platform Security Services (OPSS)
- Audit Services (IAU)
- Audit Services Append (IAU_Append)

- Audit Services Viewer (IAU_Viewer)
- WebLogic Services (WLS)
- Common Infrastructure Service (STB)

The schema `Common Infrastructure Services` 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*.

> **Note:**
>
> You must invoke RCU twice. When you invoke RCU the first time select **Database type** as **Oracle Database enabled for edition-based redefinition** and load the EBR dependent components (STB, OPSS, IAU, IAU_Append, IAU_Viewer, and WLS). When you invoke RCU the second time, select **Database type** as **Oracle Database**, provide the prefix used the first time, and select **Oracle Internet Directory** only.

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 Autonomous Transaction Processing-Dedicated (ATP-D) and Autonomous Transaction Processing-Dedicated (ATP-D)), 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.

For an Autonomous Transaction Processing-Dedicated (ATP-D) 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.

When you reach the Completion Summary screen, click **Close** to dismiss the RCU.

> **Note:**
>
> If you encounter any issues when you create schemas on an Autonomous Transaction Processing database (both Autonomous Transaction Processing-Dedicated (ATP-D) and Autonomous Transaction Processing-Dedicated (ATP-D)), 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.

> **Note:**
>
> For an Autonomous Transaction Processing Shared (ATP-S) database, before you start the Configuration Wizard, you must set the `TNS_ADMIN` property using the following command:
>
> `export TNS_ADMIN=/<$ORACLE_HOME>/network/admin.`
>
> You must change `$ORACLE_HOME` to your Oracle Home location. For example: `export TNS_ADMIN=/users/test/network/admin`
>
> Where, `/users/test/` is the Oracle Home location.

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 14c (14.1.2.1.0) 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.

> **Note:**
>
> Apply the one-off ADF patch (search for Bug ID 37376076 at https://support.oracle.com) manually using OPatch to the Oracle Internet Directory 14c (14.1.2.1.0) `ORACLE_HOME` after installation and before domain creation. This applies only to collocated OID installations and does not apply to standalone Oracle Internet Directory 14c (14.1.2.1.0) installations.

Selecting the Configuration 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 Internet Directory

On the Templates screen, make sure **Create Domain Using Product Templates** is selected, then select the following templates:

For standalone mode, select the following template:

- Oracle Internet Directory (Standalone) - [oid]

For collocated mode, select the following templates:

- Oracle Internet Directory (Collocated) - [oid]

  Selecting this template automatically selects the following as dependencies:

  - Oracle Directory Services Manager - [oid]
  - Oracle JRF - [oracle_common]
  - WebLogic Coherence Cluster Extension - [wlserver]
  - Oracle Enterprise Manager - [em]
- Oracle Directory Integration Platform - [dip]

  Optional. Select this template if you're using OID and ODIP in the same domain.

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

Configuring the Administrator Account

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

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

Specifying the Domain Mode and JDK

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

On the Domain Mode and JDK screen:

- Select **Production** in the **Domain Mode** field.
- Disable secured mode for the domain by selecting the **Disable Secure Mode** check-box.
- 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 Autonomous Transaction Processing-Dedicated (ATP-D) and Autonomous Transaction Processing-Dedicated (ATP-D)), 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`

Field	Description
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 Autonomous Transaction Processing-Dedicated (ATP-D) and Autonomous Transaction Processing-Dedicated (ATP-D)), 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=/<$ORACLE_HOME>/network/admin
```

In the connect string, you must pass `TNS_alias` as the database name found in `tnsnames.ora`, and `TNS_ADMIN` property to `<$ORACLE_HOME>/network/admin`, which is the location of the wallet files, `ojdbc.properties`, and `tnsnames.ora`.

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

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

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

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

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.

> **Note:**
>
> If you selected standalone mode, you must use the Datasources screen to specify details about the database schemas.

For an Autonomous Transaction Processing database (both Autonomous Transaction Processing-Dedicated (ATP-D) and Autonomous Transaction Processing Shared (ATP-S)), specify the connection credentials using the **Connection URL String** option only, and enter the connect string specified in `tnsnames.ora` that is present in `/<$ORACLE_HOME>/network/admin`, which is the location of the wallet files, for your service name or `TNS_alias`.

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

```
jdbc:oracle:thin:@(DESCRIPTION=(CONNECT_TIMEOUT=120)(RETRY_COUNT=20)
(RETRY_DELAY=3)
(TRANSPORT_CONNECT_TIMEOUT=3)(ADDRESS_LIST=(LOAD_BALANCE=on)
(ADDRESS=(PROTOCOL=<protocol_name>)
(HOST=<host_name>)(PORT=<port_number>)))
(CONNECT_DATA=(SERVICE_NAME=<service_name>.atp.oraclecloud.com)))
```

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

```
jdbc:oracle:thin:@(DESCRIPTION=(CONNECT_TIMEOUT=120)=(RETRY_COUNT=20)
(RETRY_DELAY=3)(ADDRESS=(PROTOCOL=<protocol_name>)
(PORT=<port_number>)(HOST=<host_name>))
(CONNECT_DATA=(SERVICE_NAME=<service_name>.adb.oraclecloud.com))
(security=(ssl_server_cert_dn="CN=example.com,
OU=<organizational_unit>, O=<organization>, L=<city>, ST=<state>,
C=<country>")))
```

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.

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

  Select Topology to configure machines and assign the Administration Server to a machine. Note that you cannot configure the oid system component using the Configuration Wizard. The oid instance is configured after the domain configuration. See Performing the Initial Oracle Internet Directory Setup.

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 Listen Address and configure the Administration Server ports.

> **Note:**
>
> The default port values will vary depening on how you conifigured your domain. The Enable SSL Listen Port is enabled by default, but the default values may change. For a list of default values, see Port Numbers by Product and Component.

1. Provide a name for the Administration Server. The name field must not be null or empty and cannot contain any special characters.
2. 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`.
3. Verify the port settings. When the domain type is set to Production, then the **Enable SSL Listen Port** option is enabled by default. Do *not* specify any server groups for the Administration Server.

> **Note:**
>
> You can change the port values as needed, but **they must be unique**. If the same port numbers are used for different ports, you will not be able to navigate to the next step in the Configuration Wizard.

For more information, see Specifying the Listen Address in *Creating WebLogic Domains Using the Configuration Wizard*.

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 information about Node Manager types, see About Node Manager in *Administering Node Manager for Oracle WebLogic Server*.

Configuring Managed Servers

If you do not plan to create a WebLogic managed server during installation, click **Next** and proceed. A WebLogic managed server is not required for OID and Oracle Directory Services Manager (ODSM) gets deployed on the administration server.

> **Note:**
>
> If you are configuring Oracle Internet Directory and Oracle Directory Integration Platform in the same domain then you must configure the Managed Server. By default, `wls_ods1` is the Managed Server for Oracle Directory Integration Platform.

If you plan to create a WebLogic managed server during installation, ensure that you associate the Server Groups to the managed server. This step deploys the ODSM/`oiddms` on the administration server.

> **Note:**
>
> Server Groups are WebLogic Server constructs that are used to organize resources such as hostname(s) being part of a 'machine'.

If you do not select any server groups for the managed server and ODSM/`oiddms` are deployed on the managed server, then use the Administration Server Console to remove `oiddms` from the managed sever and deploy them on the administration server.

Configuring a Cluster

You can skip this screen as it is not applicable to Oracle Internet Directory.

Click **Next**.

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

Defining Server Templates

Click **Next** and proceed, as this is not applicable to Oracle Internet Directory.

Configuring Coherence Clusters

You can skip this screen as it is not applicable to Oracle Internet Directory.

Click **Next**.

Creating a New Oracle Internet Directory Machine

Use the Machines screen to update the default machine listed on the screen — `oidhost1`. 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*.

Select the default machine `oidhost1` that is listed, and update the **Listen Port** to appropriate value based on the Node Manager listen port number.

> **Note:**
>
> Do not change the name of the default machine (`oidhost1`), as the WLST command `oid_setup()` run for setting up the OID instance, later during the post-configuration stage (as described in Performing the Initial Oracle Internet Directory Setup), relies on this name.

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

Assigning Servers to Oracle Internet Directory Machines

Use the Assign Servers to Machines screen to assign the Administration Server to the default machine `oidhost1` that is listed.

On the Assign Servers to Machines screen:

1. In the Machines pane, select the default machine **oidhost1** that is listed.
2. In the Servers pane, assign `AdminServer` to `oidhost1` 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 (`oidhost1`) in the Machines pane.
   - Double-click on `AdminServer` to move it beneath the selected machine (`oidhost1`) in the Machines pane.

Virtual Targets

You can skip this screen for Oracle Internet Directory configuration.

Click **Next** and proceed.

Partitions

Click **Next** as this is not applicable to Oracle Internet Directory.

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

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.

Prerequisites for an Autonomous Transaction Processing-Dedicated (ATP-D) database

In case of a standalone and collocated Oracle Internet Directory (OID) configuration, after configuring the domain, you must modify the wallet settings and update the classpath before you start the servers.

Refer to the following topics based on your configuration mode:

Prerequisites for Standalone Oracle Internet Directory Configuration with an Autonomous Transaction Processing-Dedicated (ATP-D) database

In case of a standalone Oracle Internet Directory (OID) configuration, after configuring the domain, you must modify the wallet settings and update the classpath before you start the Node Manager.

1. Copy the wallet files from `<$ORACLE_HOME>/network/admin` to `<$DOMAIN_HOME>/config/fmwconfig/components/OID/config` .
2. Update the `ojdbc.properties` file as follows:

```
# Connection property while using Oracle wallets.
#oracle.net.wallet_location=(SOURCE=(METHOD=FILE)(METHOD_DATA=(DIRECTORY=${TNS
_ADMIN})))
SSL_SERVER_DN_MATCH=yes
# FOLLOW THESE STEPS FOR USING JAVA KEYSTORE (JKS)
# (1) Uncomment the following properties to use JKS.
# (2) Comment out the oracle.net.wallet_location property above
# (3) Set the correct password for both trustStorePassword and
keyStorePassword.
# The keyStorePassword and trustStorePassword are the passwords you
specified when downloading the wallet from OCI Console
 or the Service Console..
javax.net.ssl.trustStoreType=JKS
javax.net.ssl.trustStore=<DOMAIN_HOME>/config/fmwconfig/components/OID/
config/truststor
e.jks
javax.net.ssl.trustStorePassword=<trustStorePassword>
javax.net.ssl.keyStoreType=JKS
javax.net.ssl.keyStore=<DOMAIN_HOME>/config/fmwconfig/components/OID/
config/keystore.jk
s
javax.net.ssl.keyStorePassword=<keyStorePassword>
```

> **Note:**
>
> Make sure to comment the wallet related property in `ojdbc.properties`
> For example:
>
> ```
> #oracle.net.wallet_location=(SOURCE=(METHOD=FILE)
> (METHOD_DATA=(DIRECTORY=${TNS_ADMIN})))
> ```

3. Create the file `ojdbc_OIDDB.properties` in the wallet location, `<DOMAIN_HOME>/config/fmwconfig/components/OID/config/`, and copy contents of `ojdbc.properties` to the new file `ojdbc_OIDDB.properties`.

4. Modify the wallet location in the `sqlnet.ora` file as follows:

```
WALLET_LOCATION = (SOURCE=(METHOD=FILE)
(METHOD_DATA=(DIRECTORY="<DOMAIN_HOME>/config/fmwconfig/components/OID/
config/")))
```

5. Replace all contents of `tnsnames.ora` in `<$DOMAIN_HOME>/config/fmwconfig/components/OID/config` as follows:

```
OIDDB=<connect string given in RCU>
```

See Connection Credentials for an Autonomous Transaction Processing Database.

For example:

```
OIDDB=(DESCRIPTION=(CONNECT_TIMEOUT=120)=(RETRY_COUNT=20)(RETRY_DELAY=3)
(ADDRESS=(PROTOCOL=<protocol_name>)
(PORT=<port_number>)(HOST=<host_name>))
(CONNECT_DATA=(SERVICE_NAME=<service_name>.adb.oraclecloud.com))
(security=(ssl_server_cert_dn="CN=adwc.uscom-east-1.oraclecloud.com,
OU=Oracle BMCS US, O=Oracle Corporation, L=Redwood City, ST=California,
C=US")))
```

6. Update the classpath in `<$DOMAIN_HOME>/bin/startNodeManager.sh`.
The classpath before update looks similar to:

```
POST_CLASSPATH="/home/opc/idm/mwoc5/oid/../jdbc/lib/ojdbc7_g.jar$
{CLASSPATHSEP}${POST_CLASSPATH}"
```

The classpath after update looks similar to:

```
PRE_CLASSPATH="<ORACLE_HOME>/oracle_common/modules/oracle.jdbc/ojdbc8.jar"
export PRE_CLASSPATH
POST_CLASSPATH="<ORACLE_HOME>/oracle_common/modules/oracle.jdbc/ojdbc8.jar$
{CLASSPATHSEP}${POST_CLASSPATH}"
export POST_CLASSPATH
```

Prerequisites for Collocated Oracle Internet Directory Configuration with an Autonomous Transaction Processing-Dedicated (ATP-D) database

In case of a collocated Oracle Internet Directory (OID) configuration, after configuring the domain, you must modify the wallet settings before you start the Administration Server and the Node Manager.

1. Copy the wallet files from `<$ORACLE_HOME>/network/admin` to `<$DOMAIN_HOME>/config/fmwconfig/components/OID/config` .
2. Replace all contents of `tnsnames.ora` in `<$DOMAIN_HOME>/config/fmwconfig/components/OID/config` as follows:

```
OIDDB=<connect string given in RCU>
```

See Connection Credentials for an Autonomous Transaction Processing Database.

For example:

```
OIDDB=(DESCRIPTION=(CONNECT_TIMEOUT=120)=(RETRY_COUNT=20)(RETRY_DELAY=3)
(ADDRESS=(PROTOCOL=<protocol_name>)
(PORT=<port_number>)(HOST=<host_name>))
(CONNECT_DATA=(SERVICE_NAME=<service_name>.adb.oraclecloud.com))
(security=(ssl_server_cert_dn="CN=adwc.uscom-east-1.oraclecloud.com,
OU=Oracle BMCS US, O=Oracle Corporation, L=Redwood City, ST=California,
C=US")))
```

3. Modify the wallet location in the `sqlnet.ora` file as follows:

```
WALLET_LOCATION = (SOURCE=(METHOD=FILE)
(METHOD_DATA=(DIRECTORY="<DOMAIN_HOME>/config/fmwconfig/components/OID/
config/")))
```

Starting Servers and Processes

After configuration is complete, start the servers and the processes.

For more information on additional tools you can use to manage your domain, see Overview of Oracle Fusion Middleware Administration Tools in *Administering Oracle Fusion Middleware*.

Refer to the following topics based on your configuration mode:

Starting the Servers for Standalone Oracle Internet Directory

In case of a standalone Oracle Internet Directory (OID) configuration, start the Node Manager. The OID instance will be started when you perform the initial OID setup in the later sections.

For an Autonomous Transaction Processing-Dedicated (ATP-D) database, you must modify the wallet settings and update the classpath before you start the Node Manager. See Prerequisites for Standalone Oracle Internet Directory Configuration with an Autonomous Transaction Processing-Dedicated (ATP-D) database.

To start the Node Manager, use the following command:

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

> **Note:**
>
> Before starting the Node Manager, make sure that any changes made to the default port in `nodemanager.properties` reflects in the corresponding associated machine as well.
>
> For an Autonomous Transaction Processing-Dedicated (ATP-D) database, before starting the Node Manager, set `TNS_ADMIN` property to `<$DOMAIN_HOME>/config/fmwconfig/components/OID/config/` using the following command:
>
> ```
> export TNS_ADMIN=<$DOMAIN_HOME>/config/fmwconfig/components/OID/config.
> ```

Starting Servers and Processes for Collocated Oracle Internet Directory

In case of a collocated Oracle Internet Directory (OID) configuration, start the Administration Server and the Node Manager. The OID instance will be started when you perform the initial OID setup in the later sections.

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 processes using the WLST command-line utility or a script. You can also use the Oracle Fusion Middleware Control and the Oracle WebLogic Server Remote Console. See Starting and Stopping Administration and Managed Servers and Node Manager.
>
> As of release 14c (14.1.2.0.0), the WebLogic Server Administration Console has been removed. For comparable functionality, you should use the WebLogic Remote Console. For more information, see Oracle WebLogic Remote Console.

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

> **Note:**
>
> Depending on your existing security settings, you may need to perform additional configuration before you can manage a domain with secured production mode enabled. For more information, see Connecting to the Administration Server using WebLogic Remote Console
>
> .

Step 1: Start the Administration Server

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

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

> **Note:**
>
> When using secured production mode, you must provide additional parameters to start the Administration Server. See Connecting to the Administration Server using WLST in *Administering Security for Oracle WebLogic Server*.

When prompted, enter your user name, password, and the URL of the Administration Server.

Step 2: Start Node Manager

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

- (UNIX) *`DOMAIN_HOME`*`/bin/startNodeManager.sh`

- (Windows) *DOMAIN_HOME*\bin\startNodeManager.cmd

> **Note:**
>
> Before starting the Node Manager, make sure that any changes made to the default port in `nodemanager.properties` reflects in the corresponding associated machine as well.

Step 3: Start System Components

To start system components, use the `startComponent` script:

- (UNIX) *NEW_DOMAIN_HOME*/bin/startComponent.sh *component_name*
- (Windows) *NEW_DOMAIN_HOME*\bin\startComponent.cmd *component_name*

You can start system components in any order.

Performing the Initial Oracle Internet Directory Setup

Use the `wlst` command from a different terminal to connect to Administration Server and set up Oracle Internet Directory.

To perform the initial setup of OID, do the following:

1. If you are running in secure production mode, export the following before launching the WLST tool:

   ```
   setenv WLST_PROPERTIES
   "-Dweblogic.security.TrustKeyStore=CustomTrust
            -
   Dweblogic.security.CustomTrustKeyStoreFileName=<Location_of_PKCS12_keystore
   s>/trust.p12
            -
   Dweblogic.security.CustomTrustKeyStorePassPhrase=trustKeyStorePassword"
   ```

2. Run the following command from the location *ORACLE_HOME*/oracle_common/common/bin to launch the WLST tool:

   ```
   ./wlst.sh
   ```

3. In case of a standalone Oracle Internet Directory configuration, connect to the Node Manager using the following command:

   ```
   nmConnect(username='wls_user',password='password',domainName='base_domain')
   ```

   In case of a collocated Oracle Internet Directory configuration, connect to the Administration Server using the following command:

   ```
   connect('Admin_username','Admin_password','t3://Admin_host:Admin_port')
   ```

4. Run the following command to perform the initial setup of OID:

   From location:

   - For standalone mode: `/base_domain`
   - For collocated mode: `/base_domain/serverConfig`

```
oid_setup(orcladminPassword='password',odsPassword='password',realmDN='<your realm>',port='nnnn', sslPort='nnnn', host='hostname')
```

where,

```
realmDN='<dc=<xxxx>,dc=<company name>, dc=com>'
```

> **Note:**
>
> For information about the other optional arguments that can be used with `oid_setup` command, run the following command:
>
> ```
> help('oid_setup')
> ```
>
> You can use the appropriate arguments for running OID on custom SSL and non-SSL ports, setting instanceName, port, hostname, machineName as input parameters etc.

The command `oid_setup()` performs the following operations:

- Sets the password for `cn=orcladmin user`.
- Creates the first `oid1` instance. The following parameters are set by default when `oid_setup` is run:
  - `instanceName = 'oid1'`
  - `host = 'hostname of the current machine'`
  - `port = '3060'`
  - `machine = 'oidhost1'`

    This gets created automatically when you run `config.sh`.
  - `sslPort = '3131'`
- Starts the OID instance `oid1` .
- Creates the realm.

> **Note:**
>
> If the realm is not provided then `'dc=us,dc=oracle,dc=com'` realm is created automatically.

> **Note:**
>
> For more information about managing Oracle Internet Directory components using WLST commands, see Managing Oracle Internet Directory Components by Using WLST Commands in the *Administering Oracle Internet Directory*.

Verifying the Configuration

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

To verify the Oracle Internet Directory (OID) is configured successfully, do the following:

1. Set the environment variable `ORACLE_HOME` to the new 14*c* ORACLE_HOME location.
2. Run the following command to check on the OID instance:

   ```
   ORACLE_HOME/bin/ldapbind -h OID_HOST -p OID_PORT
   ```

For additional configuration and administration tasks, see Performing Additional Domain Configuration Tasks.

5

Configuring Oracle Directory Integration Platform

Configure Oracle Directory Integration Platform (ODIP) after you install Oracle Internet Directory binaries.

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

- Preparing to Install and Configure Oracle Internet Directory
- Installing the Oracle Internet Directory Software

> **Note:**
>
> Ensure that you install Oracle Fusion Middleware Infrastructure too. Installation of ODIP requires Infrastructure to be installed.

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 Directory Integration Platform (ODIP) domain, you must install required schemas on a certified database for use with this release of Oracle Fusion Middleware.

> **Note:**
>
> You can skip this section if OID is configured as a backend directory and you've already created a schema for OID collocated mode as described in Creating 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 Autonomous Transaction Processing-Dedicated (ATP-D) and 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 14c (14.1.2.1.0), the certified JDK is 17.0.12 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.
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.

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 Autonomous Transaction Processing-Dedicated (ATP-D) and 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: As of Oracle Fusion Middleware 14c (14.1.2.1.0), new schemas are created with editions-based redefinition (EBR) views enabled by default. Oracle Internet Directory schemas do not support EBR, therefore, in order to use the EBR functionality with your non-OAM schemas (such as SOA), you will have to run the RCU twice. The first time that RCU is run, select Oracle EBR Database for the non-OAM schemas. The second time you run RCU, select Oracle Database for your Oracle Internet Directory schemas.

To provide the database connection details:

1. On the Database Connection Details screen, provide the database connection details. You have two options when creating schemas:
   - Creating schemas for components that support EBR (SOA)
   - Creating schemas for components that do not support EBR (OIM)

   For example, when you are creating schemas for components that support EBR:

   Database Type: `Oracle EBR 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`

Username: `sys`
Password: `******`
Role: `SYSDBA`

When you are creating schemas for components that do not support EBR, select `Oracle Database` as the Database Type.

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

For information about specifying connection credentials when connecting to an Oracle database, see Connection Credentials for Oracle Databases and Oracle Databases with Edition-Based Redefinition.

Specifying a Custom Prefix and Selecting Schemas

Select **Create new prefix**, specify a custom prefix, then select the **Oracle Internet Directory** schema. This action automatically selects the following schemas as dependencies:

- ODS — Select this schema only if ODIP needs to be wired against OID backend directory installed in same domain.
- Oracle Platform Security Services
- Audit Services
- Audit Services Append
- Audit Services Viewer
- WebLogic Services

The schema Common Infrastructure Services is also 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 Autonomous Transaction Processing-Dedicated (ATP-D) and 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 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 Autonomous Transaction Processing-Dedicated (ATP-D) and 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 Oracle Directory Integration Platform with Backend Directories

Oracle Directory Integration Platform (ODIP) can be configured with the Oracle Internet Directory (OID), Oracle Unified Directory (OUD), or Oracle Directory Server Enterprise Edition (ODSEE).

> **Note:**
>
> When configuring ODIP with backend directories, you must set the environment variable *ORACLE_HOME* for ODIP, to the top level Oracle home, wherever required.
>
> For example, for Oracle Internet Directory or Infrastructure installation, if wlserver is installed under `/home/Oracle/Middleware/Oracle_Home`, then *ORACLE_HOME* must be set to `/home/Oracle/Middleware/Oracle_Home`.

- To configure Oracle Directory Integration Platform with Oracle Internet Directory, see Configuring Oracle Internet Directory in the *Oracle Fusion Middleware Administrator's Guide for Oracle Directory Integration Platform*.
- To configure Oracle Directory Integration Platform with Oracle Unified Directory, see Configuring Oracle Directory Integration Platform for Oracle Unified Directory in the *Oracle Fusion Middleware Administrator's Guide for Oracle Directory Integration Platform*.
- To configure Oracle Directory Integration Platform with Oracle Directory Server Enterprise Edition, see Configuring Oracle Directory Integration Platform for Oracle Directory Server Enterprise Edition in the *Oracle Fusion Middleware Administrator's Guide for Oracle Directory Integration Platform*.

Installing ODIP Without a Database

You can install and configure Oracle Directory Integration Platform (ODIP) to run without a database.

To configure Oracle Directory Integration Platform (ODIP) to work without creating and using a database, create the following Python script, `oudscript.py`, which creates a domain for ODIP without a database. Note: replace password in the script with your WebLogic password. This sample assumes `/oracle/mw_oud14c` as the Oracle Unified Directory home. Be sure to use the directory information that matches your installation.

```
setTopologyProfile('Compact')
selectTemplate('Basic WebLogic Server Domain')
selectTemplate('Oracle Directory Integration Platform')
loadTemplates()
setOption('AppDir', '${MW_HOME}/applications/dip1')
cd(r'/Security/base_domain/User/weblogic')
cmo.setPassword(xxxxxx)
writeDomain('${MW_HOME}/domains/dip1')
closeTemplate()
readDomain('${MW_HOME}/domains/dip1')
cd('Servers/AdminServer')
```

```
cmo.setListenPort(7007)
cmo.setListenAddress('')
create('AdminServer','SSL')
cd('SSL/AdminServer')
cmo.setEnabled(true)
cmo.setListenPort(7008)
cd('/Servers/wls_ods1')
cmo.setListenPort(7009)
create('wls_ods1','SSL')
cd('SSL/wls_ods1')
cmo.setEnabled(true)
cmo.setListenPort(7010)
updateDomain()
closeDomain()
```

You can deploy this with `wlst.sh` by running the command `wlst.sh oudscript.py`. After running the script, use the dipConfigurator to configure ODIP. See Configuring Oracle Internet Directory in *Administering Oracle Directory Integration Platform*.

6

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.

Table 6-1 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. **Note:** The WebLogic Server Administration Console has been removed. For comparable functionality, you will use the WebLogic Remote Console.	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*.

Performing Additional Domain Configuration Tasks

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

Table 6-2 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 6-3 provides a list of tasks to perform if you want to scale out your standard installation environment for high availability.

Table 6-3 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 *High Availability Guide*.

Table 6-3 (Cont.) Tasks Required to Prepare Your Environment for High Availability

Task	Description	More Information
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 *HTTP Server Administration Guide*.
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 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*.

7

Configuring High Availability for Oracle Directory Services Components

This chapter describes configuring Oracle Directory Services products for high availability in an active-active configuration.

About the 14c (14.1.2.1.0) Oracle Directory Services Products

The following table summarizes Oracle Identity Management products that you can install using the suite-level installation program for 14c (14.1.2.1.0).

Table 7-1 The 14c (14.1.2.1.0) Identity Management Components and Product Suites

Product	Description	Product Suite
Oracle Internet Directory	LDAP Version 3-enabled service that enables fast retrieval and centralized management of information about dispersed users, network configuration, and other resources.	Oracle Identity Management Platform and Directory Services Suite
Oracle Directory Integration Platform	Oracle Directory Integration Platform is a J2EE application that enables you to synchronize data between various directories and the back-end directory. Oracle Directory Integration Platform includes services and interfaces that enable you to deploy synchronization solutions with other enterprise repositories.	Oracle Identity Management Platform and Directory Services Suite
Oracle Directory Services Manager	GUI for Oracle Internet Directory. Oracle Directory Services Manager that simplifies administration and configuration of Oracle Internet Directory by enabling you to use web-based forms and templates. Oracle Directory Services Manager is available from either the Oracle Enterprise Manager Fusion Middleware Control or from its own URL.	Oracle Identity Management Platform and Directory Services Suite

For more information on Oracle Internet Directory installation, See Preparing to Install and Configure Oracle Internet Directory in *Oracle Fusion Middleware Installing and Configuring Oracle Internet Directory*

Prerequisites for Oracle Directory Services High Availability Configuration

This section describes the prerequisite steps that you must complete before setting up an Oracle Directory Services high availability configuration.

Oracle Home Requirement

The Oracle home for the Identity Management components must be the same across all nodes.

```
/u01/app/oracle/product/fmw/idm
```

```
/u01/app/oracle/product/fmw/idm
```

Database Prerequisites

Several Oracle Identity Management components require the presence of a supported database and schemas.

To check if your database is certified or to see all certified databases, see the "Certified Databases" section in the Certification Document: http://www.oracle.com/technology/software/products/ias/files/fusion_certification.html.

To determine the database version, run this query:

```
SQL>select version from sys.product_component_version where product like 'Oracle%'
```

About Installing and Configuring the Database Repository

Oracle recommends a highly available database to store the metadata repository.

For maximum availability, Oracle recommends using an Oracle Real Application Clusters (Oracle RAC) database. Oracle recommends that the database use Oracle Automatic Storage Management for data storage. If you use Oracle ASM, the best practice is to also use Oracle Managed Files.

If you use Oracle ASM, install it in its own Oracle Home and have two disk groups:

- One for the Database files.
- One for the Flash Recovery Area.

Oracle Clusterware

See *Oracle Real Application Clusters Installation Guide for Linux and UNIX*.

Automatic Storage Management

See *Oracle Real Application Clusters Installation Guide for Linux and UNIX*.

When you run the installer, select Configure Automatic Storage Management in the Select Configuration page to create a separate Automatic Storage Management home.

Oracle Real Application Clusters

See *Oracle Real Application Clusters Installation Guide for Linux and UNIX*.

Many Oracle Fusion Middleware components require that schemas are in a database prior to installation. Use the Repository Creation Utility (RCU) to create the component schemas in an existing database. For high availability environments, you must create the schemas and load them into an Oracle RAC database.

Configuring the Database for Oracle Fusion Middleware Metadata

You need to have the network prerequisites for deploying an Oracle Identity Management high availability environment.

Create the Oracle Real Application Clusters database to store Oracle Fusion Middleware 14c (14.1.2.1.0) metadata with the following characteristics:

- It should be in archive log mode to facilitate backup and recovery.
- Optionally, flashback should be enabled.
- It should be created with the ALT32UTF8 character set.

The value of the static PROCESSES initialization parameter must be 500 or greater for Oracle Internet Directory. This value is checked by the Repository Creation Utility.

To check the value, you can use the SHOW PARAMETER command in SQL*Plus:

```
prompt> sqlplus "sys/password as sysdba"
SQL> SHOW PARAMETER processes
```

One common way to change the parameter value is to use a command similar to the following and then stop and restart the database to make the parameter take effect:

```
prompt> sqlplus "sys/password as sysdba"
SQL> ALTER SYSTEM SET PROCESSES=500 SCOPE=SPFILE;
```

The method that you use to change a parameter's value depends on whether the parameter is static or dynamic, and on whether your database uses a parameter file or a server parameter file.

See:

Database Examples in this Chapter

See the databases used in Oracle Directory Services Configuration examples in this chapter.

Table 7-2 Databases Used in Identity Management Configuration Examples

Component	Database Service Name	Database Instance Name
Oracle Internet Directory	oid.example.com	oiddb1, oiddb2
Oracle Directory Integration Platform	oid.example.com	oiddb1, oiddb2
Oracle Directory Services Manager	N/A	N/A

Configuring Database Services

Oracle recommends using the Oracle Enterprise Manager Cluster Managed Services Page to create database services that client applications use to connect to the database.

You can also use SQL*Plus to configure your Oracle RAC database to automate failover for Oracle Internet Directory using the following instructions. Note that each of the following commands has to be run on only one node in the cluster:

1. Use the CREATE_SERVICE subprogram to both create the database service and enable high availability notification and configure server-side Transparent Application Failover (TAF) settings.

   ```
   prompt> sqlplus "sys/password as sysdba"

   SQL> EXECUTE DBMS_SERVICE.CREATE_SERVICE
   (SERVICE_NAME => 'idm.example.com',
   NETWORK_NAME => 'idm.example.com',
   AQ_HA_NOTIFICATIONS => TRUE,
   FAILOVER_METHOD => DBMS_SERVICE.FAILOVER_METHOD_BASIC,
   FAILOVER_TYPE => DBMS_SERVICE.FAILOVER_TYPE_SELECT,
   FAILOVER_RETRIES => 5, FAILOVER_DELAY => 5);
   ```

   You must enter the EXECUTE DBMS_SERVICE command on a single line.

2. Add the service to the database and assign it to the instances using `srvctl`.

   ```
   prompt> srvctl add service -d idmdb -s idm -r idmdb1,idmdb2
   ```

3. Start the service using `srvctl`.

   ```
   prompt> srvctl start service -d idmdb -s  idm
   ```

If you already have a service in the database, ensure that it is enabled for high availability notifications and configured with the proper server-side Transparent Application Failover (TAF) settings. Use the DBMS_SERVICE package to modify the service to enable high availability notification to go through Advanced Queuing (AQ) by setting the AQ_HA_NOTIFICATIONS attribute to TRUE and configure server-side TAF settings, as shown below:

```
prompt> sqlplus "sys/password as sysdba"

SQL> EXECUTE DBMS_SERVICE.MODIFY_SERVICE
(SERVICE_NAME => 'idm.example.com',
AQ_HA_NOTIFICATIONS => TRUE,
FAILOVER_METHOD => DBMS_SERVICE.FAILOVER_METHOD_BASIC,
FAILOVER_TYPE => DBMS_SERVICE.FAILOVER_TYPE_SELECT,
FAILOVER_RETRIES => 5, FAILOVER_DELAY => 5);
```

You must enter the EXECUTE DBMS_SERVICE command on a single line.

See Also:

- Administering Services with Oracle Enterprise Manager, PL/SQL, and SRVCTL in *Oracle Real Application Clusters Administration and Deployment Guide*
- DBMS_SERVICE in *Oracle Database PL/SQL Packages and Types Reference*

Verifying Transparent Application Failover

After the Oracle Internet Directory process starts, you can query the FAILOVER_TYPE, FAILOVER_METHOD, and FAILED_OVER columns in the V$SESSION_VIEW to obtain information about connected clients and their TAF status.

For example, use the following SQL statement to verify that TAF is correctly configured:

```
SELECT MACHINE, FAILOVER_TYPE, FAILOVER_METHOD, FAILED_OVER, COUNT(*)
FROM V$SESSION
GROUP BY MACHINE, FAILOVER_TYPE, FAILOVER_METHOD, FAILED_OVER;
```

The output before failover is similar to this:

```
MACHINE              FAILOVER_TYPE FAILOVER_M  FAI   COUNT(*)
-------------------- ------------- ---------- ---- ----------
oidhost1             SELECT        BASIC      NO           11
oidhost1             SELECT        BASIC      NO            1
```

The output after failover is similar to this:

```
MACHINE              FAILOVER_TYPE FAILOVER_M  FAI   COUNT(*)
-------------------- ------------- ---------- ---- ----------
oidhost2             SELECT        BASIC      NO           11
oidhost2             SELECT        BASIC      NO            1
```

Configuring Virtual Server Names and Ports for the Load Balancer

There are network prerequisites for Load Balancer and Virtual Server Names for deploying an Oracle Identity Management high availability environment.

Load Balancers

All components in the Oracle Identity Management software stack require a hardware load balancer when deployed in a high availability configuration.

The hardware load balancer should have the following features:

- Ability to load-balance traffic to a pool of real servers through a virtual hostname: Clients access services using the virtual hostname (instead of using actual host names). The load balancer can then load balance requests to the servers in the pool.
- Port translation configuration: The load balancer should have the ability to perform port translation, where it enables incoming requests received on one port to be routed to a server process running on a different port. For example, a request received on port 80 can be routed to port 7777.
- Protocol translation: The load balancer should support protocol translation between systems running different protocols. It enables users on one network to access hosts on another network, despite differences in the native protocol stacks associated with the originating device and the targeted host. For example, incoming requests can be HTTPS, and outgoing requests can be HTTP.

  This feature is recommended but not required.
- SSL acceleration: SSL acceleration is a method of offloading the processor-intensive public key encryption algorithms involved in SSL transactions to a hardware accelerator.

  This feature is recommended but not required.
- Monitoring of ports (HTTP, HTTPS, LDAP, LDAPS)

- Virtual servers and port configuration. Ability to configure virtual server names and ports on your external load balancer, and the virtual server names and ports must meet the following requirements:

  The load balancer should enable configuration of multiple virtual servers. For each virtual server, the load balancer should enable configuration of traffic management on more than one port. For example, for Oracle Internet Directory clusters, the load balancer needs to be configured with a virtual server and ports for LDAP and LDAPS traffic.

  The virtual server names must be associated with IP addresses and be part of your DNS. Clients must be able to access the load balancer through the virtual server names.

- Ability to detect node failures and immediately stop routing traffic to the failed node.
- Resource monitoring / port monitoring / process failure detection.

  The load balancer must be able to detect service and node failures (through notification or some other means) and to stop directing non-Oracle Net traffic to the failed node. If your load balancer has the ability to automatically detect failures, you should use it.

- Fault-tolerant mode

  It is highly recommended that you configure the load balancer to be in fault-tolerant mode.

- Other

  Oracle recommends that you configure the load balancer virtual server to return immediately to the calling client when the back-end services that it forwards traffic to are unavailable. This is preferred over the client disconnecting on its own after a timeout based on the TCP/IP settings on the client machine.

- Sticky routing capability

  Ability to maintain sticky connections to components based on cookies or URL.

The following table shows the virtual server names to use for the external load balancer in the Oracle Identity Management high availability environment.

Table 7-3 Virtual Server Names for the External Load Balancer

Component	Virtual Server Name
Oracle Internet Directory	oid.example.com
Oracle Directory Services Manager Console	admin.example.com

Virtual Server Names

You should setup virtual server names for the high availability deployments. Ensure that the virtual server names are associated with IP addresses and are part of your Domain Name System (DNS). The computers on which Oracle Fusion Middleware is running must be able to resolve these virtual server names.

oid.example.com

This virtual server acts as the access point for all LDAP traffic to the Oracle Internet Directory servers in the directory tier. Traffic to both the SSL and non-SSL ports is configured. The clients access this service using the address oid.example.com:636 for SSL and oid.example.com:389 for non-SSL.

Monitor the heartbeat of the Oracle Internet Directory processes on OIDHOST1 and OIDHOST2. If an Oracle Internet Directory process stops on OIDHOST1 or OIDHOST2, or if

either host is down, the load balancer must continue to route the LDAP traffic to the surviving computer.

Oracle Internet Directory High Availability

This section provides an introduction to Oracle Internet Directory and describes how to design and deploy a high availability environment for Oracle Internet Directory.

About Oracle Internet Directory Component Architecture

Oracle Internet Directory is an LDAP store that can be used by Oracle components such as Directory Integration Platform, Oracle Directory Services Manager, JPS, and also by non-Oracle components. These components connect to Oracle Internet Directory using the LDAP or LDAPS protocols.

The Oracle directory replication server uses LDAP to communicate with an Oracle directory (LDAP) server instance. To communicate with the database, all components use OCI/Oracle Net Services. Oracle Directory Services Manager and the command-line tools communicate with the Oracle directory servers over LDAP.

An Oracle Internet Directory node consists of one or more directory server instances connected to the same directory store. The directory store—that is, the repository of the directory data—is an Oracle database.

An Oracle Internet Directory node includes the following major elements:

Table 7-4 An Oracle internet Directory Node

Element	Description
Oracle directory server instance	Also called either an LDAP server instance or a directory server instance, it services directory requests through a single Oracle Internet Directory dispatcher process listening at specific TCP/IP ports. There can be more than one directory server instance on a node, listening on different ports.
Oracle directory replication server	Also called a replication server, it tracks and sends changes to replication servers in another Oracle Internet Directory system. There can be only one replication server on a node. You can choose whether to configure the replication server. If there are multiple instances of Oracle Internet Directory that use the same database, only one of them can be running replication. This is true even if the Oracle Internet Directory instances are on different nodes. The replication sever process is a process within Oracle Internet Directory. It only runs when replication is configured. For more information on Oracle Internet Directory replication, see Configuring Identity Management for Maximum High Availability..
Oracle Database Server	Stores the directory data. Oracle strongly recommends that you dedicate a database for use by the directory. The database can reside on the same node as the directory server instances.

Table 7-4 (Cont.) An Oracle internet Directory Node

Element	Description
OID Monitor (OIDMON)	Initiates, monitors, and terminates the LDAP server and replication server processes. When you invoke process management commands, such as oidctl or Node Manager, or when you use Fusion Middleware Control to start or stop server instances, your commands are interpreted by this process. OIDMON also monitors servers and restarts them if they have stopped running for abnormal reasons. OIDMON starts a default instance of OIDLDAPD. If the default instance of OIDLDAPD is stopped using the OIDCTL command, then OIDMON stops the instance. When OIDMON is restarted by Node Manager (using `startComponent.sh`), OIDMON restarts the default instance. All OID Monitor activity is logged in the file `DOMAIN_HOME/servers/OID/logs/oid1/oidmon-xxxx.log`. This file is on the Oracle Internet Directory server file system.
OID Control Utility (OIDCTL)	Communicates with OID Monitor by placing message data in Oracle Internet Directory server tables. This message data includes configuration parameters required to run each Oracle directory server instance. Normally used from the command line only to stop and start the replication server.

Oracle Internet Directory Component Characteristics

Oracle Internet Directory, which is Oracle's LDAP store, is a C-based component that uses a database as its persistence store. It is a stateless process and stores all of the data and the majority of its configuration information in the back-end database. It uses Oracle Net Services to connect to the database.

Runtime Processes

Oracle Internet Directory has the following runtime processes:

- **OIDLDAPD**: This is the main process for Oracle Internet Directory. OIDLDAPD consists of a dispatcher process and a server process. The dispatcher process spawns the OIDLDAPD server processes during startup. Each OIDLDAPD dispatcher process has its own SSL and non-SSL ports for receiving requests. Every OID instance has one dispatcher and one server process by default. The number of server processes spawned for an instance is controlled by the `orclserverprocs` attribute.
- **OIDMON**: OIDMON is responsible for the process control of an Oracle Internet Directory instance. This process starts, stops, and monitors Oracle Internet Directory. During startup OIDMON spawns the OIDLDAPD dispatcher process and the replication server process, if replication is configured for the instance.
- **Replication server process**: This is a process within Oracle Internet Directory that runs only when replication is configured. The replication server process is spawned by OIDMON during startup.

- **Node Manager**: Node Manager is a daemon process that monitors Oracle Fusion Middleware components, including Oracle Internet Directory.

  Node Manager is responsible for the direct start, stop, restart and monitoring of OIDMON. It does not start or stop the server process directly.

Process Lifecycle

Node Manager is responsible for the direct start, stop, restart and monitoring of the daemon process, OIDMON (ORACLE_HOME/bin/oidmon). OIDMON is responsible for the process control of an Oracle Internet Directory instance.

Process Status Table

Oracle Internet Directory process information is maintained in the ODS_PROCESS_STATUS table in the ODS database user schema. OIDMON reads the contents of the table at a specified interval and acts upon the intent conveyed by the contents of that table. The interval is controlled by the value of the sleep command line argument used at OIDMON startup, and the default value is 10 seconds.

Starting and Stopping Oracle Internet Directory

An Oracle Internet Directory instance can be started and stopped using system component management scripts — `startComponent.sh` and `stopComponent.sh`.

Start Process

The start process for Oracle Internet Directory is:

1. Upon receiving the start command, Node Manager issues an oidmon start command with appropriate arguments.
2. OIDMON then starts all Oracle Internet Directory Server instances whose information in the ODS_PROCESS_STATUS table has state value 1 or 4 and COMPONENT_NAME, INSTANCE_NAME values matching the environment parameters set by Node Manager.

Stop Process

The stop process for Oracle Internet Directory is:

1. Upon receiving the stop command, Node Manager issues an oidmon stop command.
2. For each row in the ODS_PROCESS_STATUS table that matches the environment parameters COMPONENT_NAME, and INSTANCE_NAME, the oidmon stop command kills OIDMON, OIDLDAPD, and OIDREPLD processes and updates the state to 4.

Monitoring

Node Manager does not monitor server processes directly. Node Manager monitors OIDMON and OIDMON monitors the server processes. The events are:

- When you start OIDMON through Node Manager, Node Manager starts OIDMON and ensures that OIDMON is up and running.
- If OIDMON goes down for some reason, Node Manager brings it back up.
- OIDMON monitors the status of the Oracle Internet Directory dispatcher process, LDAP server processes, and replication server process and makes this status available to Node Manager.

Request Flow

Once the Oracle Internet Directory (OID) process starts up, clients access OID using the LDAP or LDAPS protocol. There is no affect on other running instances when an OID instance starts up

Oracle Internet Directory listener/dispatcher starts a configured number of server processes at startup time. The number of server processes is controlled by the orclserverprocs attribute in the instance-specific configuration entry. The default value for orclserverprocs is 1. Multiple server processes enable OID to take advantage of multiple processor systems.

The OID dispatcher process sends the LDAP connections to the OID server process in a round robin fashion. The maximum number of LDAP connections accepted by each server is 1024 by default. This number can be increased by changing the attribute orclmaxldapconns in the instance-specific configuration entry, which has a DN of the form:

```
cn=componentname,cn=osdldapd,cn=subconfigsubentry
```

Database connections from each server process are spawned at server startup time, depending on the value set for the instance configuration parameters ORCLMAXCC and ORCLPLUGINWORKERS. The number of database connections spawned by each server equals ORCLMAXCC + ORCLPLUGINWORKERS + 2. The OID server processes communicate with the Oracle database server through Oracle Net Services. An Oracle Net Services listener/dispatcher relays the request to the Oracle database. For more information, see **Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory**.

About Configuration Artifacts

The storage location requires a DB connect string. `TNSNAMES.ORA` is stored in DOMAIN_HOME/config. The wallet is stored in `DOMAIN_HOME/config/fmwconfig/components/OID/admin` (The DB ODS user password is stored in the wallet).

External Dependencies

Oracle Internet Directory uses an Oracle database to store configuration information as well as data. It uses the ODS schema to store this information.

The Oracle directory replication server uses LDAP to communicate with an Oracle directory (LDAP) server instance. To communicate with the database, all components use OCI/Oracle Net Services. Oracle Directory Services Manager and the command-line tools communicate with the Oracle directory servers over LDAP.

Oracle Internet Directory Log File

Log files for Oracle Internet Directory are under the following directory:

`DOMAIN_HOME/servers/OID/logs/`*`InstanceName`*`/`

Table shows Oracle Internet Directory processes and the log file name and location for the process.

Table 7-5 Locations of Oracle Internet Directory Process Log Files

Process	Log File Location
Directory server (oidldapd)	`DOMAIN_HOME/servers/OID/logs/InstanceName/oidldapd00sPID-XXXX.log` where: 00 is the instance number (00 by default) s stands for server PID is the server process identifier XXXX is a number from 0000 to orclmaxlogfilesconfigured. Once the orclmaxlogfilesconfigured value is reached, it starts over again from 0000. When it starts over, it truncates the file to 0 bytes. `DOMAIN_HOME/servers/OID/logs/InstanceName/oidstackInstNumberPID.log`
LDAP dispatcher (oiddispd)	`DOMAIN_HOME/servers/OID/logs/InstanceName/oiddispd00-XXXX.log` where: 00 is the instance number (00 by default) XXXX is a number from 0000 to orclmaxlogfilesconfigured
OID Monitor (OIDMON)	`DOMAIN_HOME/servers/OID/logs/InstanceName/oidmon-XXXX.log` where: XXXX is a number from 0000 to orclmaxlogfilesconfigured
Directory replication server (oidrepld)	`DOMAIN_HOME/servers/OID/logs/InstanceName/oidrepld-XXXX.log` where: XXXX is a number from 0000 to orclmaxlogfilesconfigured

For more information on using log files to troubleshoot Oracle Internet Directory, see Troubleshooting Oracle Internet Directory High Availability.

Understanding Oracle Internet Directory High Availability Concepts

This section provides conceptual information about using Oracle Internet Directory in a high availability two-node Cluster Configuration.

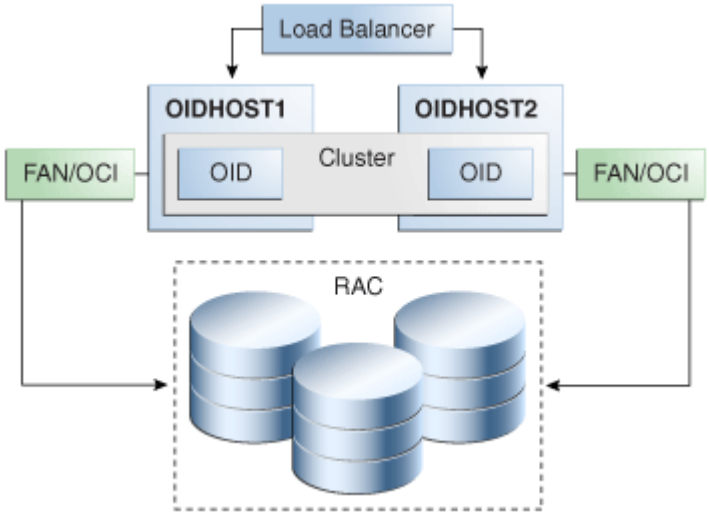
See Oracle Internet Directory Prerequisites for prerequisites and Oracle Internet Directory High Availability Configuration Steps to set up the two-node Cluster Configuration.

Oracle Internet Directory High Availability Architecture

Learn about the Oracle Internet Directory Cluster Configuration high availability architecture in an active-active configuration.

The Figure 7-1 shows the Oracle Internet Directory Cluster Configuration high availability architecture in an active-active configuration.

Figure 7-1 Oracle Internet Directory Cluster Configuration High Availability Architecture



The Figure 7-1 shows Oracle Internet Directory (OID) in the directory tier in a Cluster Configuration high availability architecture. Clustering is set up at installation time. The load balancing router routes LDAP client requests to the two OID instances that are clustered on `OIDHOST1`and `OIDHOST2`.

Transparent Application Failover (TAF) is used to connect the OID instances with the Oracle RAC database that serves as the security metadata repository. The Oracle RAC database is configured in `TNSNAMES.ORA`. High availability event notification is used for notification when an Oracle RAC instance becomes unavailable.

Starting and Stopping the Cluster

In the Cluster Configuration, Node Manager (`startComponent.sh` and `stopComponent.sh` commands) start each OID instance. There is no affect on OID at startup. A new database connection spawns when OID starts.

When the cluster is stopped using Node Manager (`stopComponent.sh` command), OID disconnects from the database and the OID server stops.

Cluster-Wide Configuration Changes (OID)

When you deploy Oracle Internet Directory in a high availability configuration, all Oracle Internet Directory instances in the cluster share the same database. Any changes made to Oracle Directory Integration Platform on one Oracle Internet Directory node automatically propagate to all the Oracle Internet Directory instances in the cluster.

Directory Synchronization Profiles

Changes that you make to directory integration profiles on one Oracle Internet Directory node do not replicate automatically to other Oracle Internet Directory nodes in a default multimaster Oracle Internet Directory replication environment. You must copy changes from the primary node to the secondary nodes manually and do so on a periodic basis. By doing this, a directory synchronization profile can run on a secondary node if a problem occurs on the primary node.

Oracle Directory Integration Platform uses the parameter `orcllastappliedchangenumber`. The value assigned to the `lastchangenumber` attribute in a directory synchronization profile

depends on the directory server on which Oracle Directory Integration Platform is running. In an active-active Oracle Directory Integration Platform configuration, you must manually update the `lastchangenumber` attribute in all instances.

To synchronize directory provisioning profiles between the primary Oracle Internet Directory node and secondary nodes:

1. On the primary node, use the `ldifwrite` command to create an LDIF dump of the entries from this container:

   ```
   cn=subscriber profiles,cn=changelog subscriber,cn=oracle internet directory
   ```

2. Copy the LDIF dump to the secondary node.
3. Use the `ldapadd` command to add the profiles on the secondary node.

After you copy an export profile to a target node, you must update the `lastchangenumber` attribute with the target node value. To update the value:

1. Disable the synchronization profile.
2. Get the value of the `lastchangenumber` attribute on the target node using the `ldapsearch` command.
3. Use `ldapsearch` to get the LDIF dump of the profile entry.
4. Use `ldapadd` to add the profile to the other Managed Server instance.
5. Go to the Oracle Directory Integration Platform Admin console and select the profile. Select **Edit**. Select the **Advanced** tab then select **Edit and Persist**. Enter the value of the `lastchangenumber` attribute. Save the profile.
6. Enable the synchronization profile.

Directory Provisioning Profiles

In a default multimaster Oracle Internet Directory replication environment, Oracle Directory Integration Platform is installed in the same location as the primary Oracle Internet Directory. The information and steps in this topic applies only when multimaster replication is set up.

If the primary node fails, event propagation stops for all profiles located on the node. Although the events are queued and not lost while the primary node is stopped, the events do not propagate to any applications that expect them. To ensure that events continue to propagate even when the primary node is down for the Version 1.0 and 2.0 profiles, the directory provisioning profiles must be copied to other secondary nodes.

However, copy directory provisioning profiles from the primary node to any secondary nodes immediately after an application is installed and before any user changes are made in Oracle Internet Directory.

To synchronize directory provisioning profiles between a primary node and any secondary nodes:

1. On the primary node, use the ldifwrite command to create an LDIF dump of the entries from this container:

   ```
   cn=provisioning profiles,cn=changelog subscriber,cn=oracle internet directory
   ```

2. Copy the LDIF dump to the secondary node.
3. Use the `ldapadd` command to add the profiles on the secondary node.

Protection from Failures and Expected Behavior

This section discusses protection from different types of failure in an OID Cluster Configuration.

Oracle Internet Directory Process Failure

OIDMON monitors OID processes. If the OID process goes down, OIDMON tries to restart it.

Node Manager monitors OIDMON. If OIDMON goes down, Node Manager restarts OIDMON.

If you cannot start an OID process, the front-ending load balancing router detects failure of OID instances in the Cluster Configuration and routes LDAP traffic to surviving instances. In case of failure, the LDAP client retries the transaction. If the instance fails in the middle of a transaction, the transaction is not committed to the database. When the failed instance comes up again, the load balancing router detects this and routes requests to all the instances.

If an OID instance in the Cluster Configuration gets hung, the load balancing router detects this and routes requests to surviving instances.

If one OID instance in the two-node Cluster Configuration fails (or if one of the computers hosting an instance fails), the load balancing router routes clients to the surviving OID instance.

Expected Client Application Behavior When Failure Occurs

Oracle Internet Directory server failure is usually transparent to OID clients as they continue to get routed through the load balancer. External load balancers are typically configured to perform a health check of OID processes. If a request is received before the load balancer detects process unavailability, clients application could receive a error. If the client application performs a retry, the load balancer should route it to a healthy OID instance and the request should be successful.

In OID active-active configurations, if you are doing ldapadd operations through the LDIF file at the time of failover, your operation would fail even if you are doing this operation through a load balancer host and port. This is because OID is down for a fraction of a second. For most applications, this will not be an issue because most applications have the ability to retry the connection a fixed number of times.

External Dependency Failure

This section describes the protection available for OID from database failure.

By default, the `tnsnames.ora` file configured in OID's ORACLE_INSTANCE ensures that OID's connections to the database are load balanced between the Oracle RAC database instances. For example, if an OID instance establishes four database connections, two connections are made to each database instance.

Oracle Internet Directory uses database high availability event notification to detect database node failure and to fail over to a surviving node.

If Transparent Application Failover (TAF) is configured, then upon a database instance failure, OID will fail over its database connections to the surviving database instance, which enables the LDAP search operations that were in progress during the failover to be continued.

If both TAF and high availability event notification are configured, TAF is used for failover and high availability event notifications are used only for logging the events. The high availability event notifications are logged in OIDLDAPD log file.

Oracle Internet Directory also has a mechanism to detect stale database connections, which enables OID to reconnect to the database.

If none of the database instances are available for a prolonged period, then the OID LDAP and REPL processes will automatically be shut down. However, OIDMON and Node Manager will continue to ping for the database instance availability and when the database becomes available, the OID processes (LDAP and REPL) are automatically restarted by OIDMON.

While all database instances are down, OIDMON continues to be up and an `oid_instanceStatus(instanceName = 'instance-name')` command shows that OIDLDAPD instances are down. When a database instance becomes available, OIDMON restarts all configured OID instances.

All database failover induced activity for OID is recorded in the OIDMON log file.

Oracle Internet Directory Prerequisites

This section describes prerequisites for setting up the OID high availability architecture.

Synchronizing the Time on Oracle Internet Directory Nodes

Before setting up OID in a high availability environment, you must ensure that the time on the individual OID nodes is synchronized.

Synchronize the time on all nodes using Greenwich Mean Time so that there is a discrepancy of no more than 250 seconds between them.

If OID Monitor detects a time discrepancy of more than 250 seconds between the two nodes, the OID Monitor on the node that is behind stops all servers on its node. To correct this problem, synchronize the time on the node that is behind in time. The OID Monitor automatically detects the change in the system time and starts the OID servers on its node.

If there are more than two nodes, the same behavior is followed. For example, assume that there are three nodes, where the first node is 150 seconds ahead of the second node, and the second node is 150 seconds ahead of the third node. In this case, the third node is 300 seconds behind the first node, so the OID Monitor will not start the servers on the third node until the time is synchronized.

Load Balancer Virtual Server Names for Oracle Internet Directory

When you deploy OID in a high availability configuration, Oracle recommends using an external load balancer to front-end OID instances and load balance requests between the OID instances.

See Configuring Virtual Server Names and Ports for the Load Balancer.

Oracle Internet Directory High Availability Configuration Steps

You can deploy Oracle Internet Directory in a High Availability configuration as part of a WebLogic Server domain.

Oracle recommends that you set up OID in a clustered deployment in which clustered OID instances access the same Oracle RAC database repository.

Installing Oracle Fusion Middleware Components

This section describes how to install the required binaries for the Oracle WebLogic Server (WL_HOME) and Oracle Home for (ORACLE_HOME) for Oracle Identity Management.

Oracle strongly recommends that you read the release notes for any additional installation and deployment considerations prior to starting the setup process.

Installing Oracle WebLogic Server

This section describes the procedure to install Oracle WebLogic server.

See Understanding Your Installation Starting Point in *Oracle Fusion Middleware Installation Planning Guide* for the Oracle WebLogic Server version to use with the latest Oracle Fusion Middleware version.

Ensure that system, patch, kernel and other requirements are met as described in Planning the Oracle WebLogic Server Installation in *Oracle Fusion Middleware Installation Guide for Oracle WebLogic Server*.

Start the Oracle WebLogic Server installer then follow these steps:

1. On the Welcome screen, click **Next**.
2. On the Choose Installation Location screen, browse and navigate to the folder where you want to install the WebLogic Servre.

   Click **Next**
3. On the Installation Type screen, Select **Fusion Middleware Insfrastructure**
4. On the Prerequisite Checks screen, Click **Next**.
5. On the Installation Summary screen, the window contains a list of the components you selected for installation, along with the approximate amount of disk space to be used by the selected components once installation is complete.

   Click **Install**.
6. On the Installation Progress, Click **Next**.
7. On the **Installation Complete** screen, click **Finish**.

Installing Oracle Internet Directory

This section describes the procedure to install Oracle Internet Directory.

Ensure that the system, patch, kernel and other requirements are met. These are listed in the Preparing to Install in *Oracle Fusion Middleware Installation Guide for Oracle Identity Management*.

> **Note:**
>
> Ensure that the ORACLE_HOME that are using for installing OID is same as ORACLE_HOME used for installing weblogic server.

On Linux platforms, if the `/etc/oraInst.loc` file exists, verify that its contents are correct. Specifically, check that the inventory directory is correct and that you have write permissions for it. If the`/etc/oraInst.loc` file does not exist, skip this step.

Start the installer for Oracle Fusion Middleware components.

Before starting the install, ensure that the following environment variables are not set:

- LD_ASSUME_KERNEL

On the Specify Inventory Directory screen, do the following:

- Enter `HOME/oraInventory`, where HOME is the home directory of the user performing the installation (this is the recommended location).
- Enter the OS group for the user performing the installation. Click **Next**.

For a UNIX install, follow the instructions on screen to run `createCentralInventory.sh` as root.

Click OK.

Proceed as follows:

1. Start Oracle Internet Directory 14c (14.1.2.1.0) Installer.
2. On the Welcome screen, click **Next**.
3. On the **Auto Updates** screen, select **Skip Auto Updates** and click **Next**.
4. On the **Installation Location** screen, browse and select the folder where you want to install Oracle Internet Directory. Click **Next**

> **Note:**
>
> Ensure that the ORACLE_HOME used for installing OID is same as ORACLE_HOME used for installing Weblogic server.

5. On the **Installation Type** screen, Based on your requirement, Select either of the option — **Standalone Oracle Internet Directory Server (Managed Independently of WebLogic server)** or **Collocated Oracle Internet Directory Server (Managed through Weblogic server)** . Click**Next**.
6. On the **JDK Selection** screen, browse and select `JDK` folder and click **Next**.
7. On the **Prerequisite Checks** ensure that all the prerequisites are met, without any warnings. Click **Next**.
8. On the **Installation Summary** screen, click **Install**.
9. Click **Finish**.

Creating Oracle Internet Directory Schemas in the Repository Using RCU

This section describes the procedure to create schemas in the repository using Repository Creation Utility (RCU).

To run RCU and create Identity Management schemas in a RAC database repository:

1. Run this command:

   ```
   ORACLE_HOME/oracle_common/bin/rcu &
   ```

2. On the Welcome screen, click **Next**.
3. On the Create Repository screen, select the **Create Repository** and **System Load and Product Load** to load component schemas into an existing database.

   Click **Next**.
4. On the Database Connection Details screen, enter connection information for the existing database as follows:
   - **Database Type:** `Oracle Database`
   - **Connection String Format**: select either —

  - Connection Parameters: This option provides an interface that accepts all connection parameters (namely - host, port and service name) separately in different UI elements.
  - Connection String: This option accepts all parameters in a single string. This string can be of one of the following formats:

    `<host>:<port>/service` **or** `<host>:<port>:<SID>` **or** `(DESCRIPTION=(ADDRESS=(host=host_name)(protocol=protocol_name)(port=port_number))(CONNECT_DATA=(SERVICE_NAME=service_name)))`

- **Host Name:**Name of the computer on which the database is running. For an Oracle RAC database, specify the VIP name or one node name. Example: `INFRADBHOST1-VIP or INFRADBHOST2-VIP`
- **Port:**The port number for the database. Example: `1521`
- **Service Name:**The service name of the database. Example: `oid.example.com`
- **Username:**`SYS`
- **Password**: The SYS user password
- **Role:**`SYSDBA`

5. Click **Next.**
6. On the **Select Components** screen, create a new prefix and select the components to be associated with this deployment:

   Create a New Prefix:`idm` (Entering a prefix is optional if you select only **Identity Management**(Oracle Internet Directory - ODS) in the **Components** field)

   Components: Select **Identity Management**(Oracle Internet Directory - ODS). On selecting `Identity Management` component, some of the default components that are dependent on Oracle Internet Directory are automatically selected.

   Click **Next**.
7. On the Schema Passwords screen, enter the passwords to create password for the main and auxiliary schema users.

   Click **Next**.
8. On the **Map Tablespaces** screen, select the tablespaces for the components. The default tablespaces for the selected components are displayed. Click **Next**
9. On the Summary screen, click **Create**.
10. On the Completion Summary screen, click **Close**.

Configuring Oracle Internet Directory With a WebLogic Domain

In this configuration, OID and a WebLogic Server domain is configured on the first host and the second host. The OID instance on the second host joins the domain created on the first host.

Oracle Internet Directory Component Names Assigned by Oracle Identity Management Installer

When you configure OID using the Config Wizard, the default instance that the installer assigns to the OID instance is `oid1`. You cannot change this name.

The instance-specific configuration entry for this OID instance is `cn=oid1, cn=osdldapd, cn=subconfigsubentry`.

If you perform a second OID installation on another computer and that OID instance uses the same database as the first instance, the installer detects the previously installed OID instance on the other computer using the same Oracle database, so it gives the second OID instance a component name of `oid2`.

The instance-specific configuration entry for the second OID instance is `cn=oid2, cn=osdldapd, cn=subconfigsubentry`. Any change of properties in the entry `cn=oid2, cn=osdldapd, cn=subconfigsubentry` will not affect the first instance `(oid1)`.

If a third OID installation is performed on another computer and that instance uses the same database as the first two instances, the installer gives the third OID instance a component name of oid3, and so on for additional instances on different hosts that use the same database.

Note that the shared configuration for all OID instances is `cn=dsaconfig, cn=configsets,cn=oracle internet directory`. Any change in this entry will affect all the instances of OID.

Configuring Oracle Internet Directory on OIDHOST1

Ensure that the schema database is running and that RCU has been used to seed the ODS database schema, then follow these steps to configure the OID instance on OIDHOST1:

1. Ensure that the system, patch, kernel and other requirements are met. These are listed in Preparing to Install in *Oracle Fusion Middleware Installation Guide for Oracle Identity Management*guide.
2. Ensure that Oracle Identity Management software is installed and upgraded on OIDHOST1 Installing Oracle Fusion Middleware Components describes.
3. Ensure that ports 3060 and 3131 are not in use by any service on the computer by issuing these commands for the operating system you are using. If a port is not in use, no output is returned from the command.

   On UNIX:

   ```
   netstat -an | grep LISTEN | grep ":3060"
   netstat -an | grep LISTEN | grep ":3131"
   ```

   On Windows:

   ```
   netstat -an | findstr "LISTEN" | findstr ":3060"
   netstat -an | findstr "LISTEN" | findstr ":3131"
   ```

4. If the port is in use (if the command returns output identifying the port), you must free the port.

   a. On Unix:

      Remove any entries for ports 3060 and 3131 in the `/etc/services` file and restart the services, or restart the computer. You can also check for any existing processes that is using these ports, using `netstat -anp` command.

   b. On Windows:

      Stop the component that is using these ports.

5. Start the Configuration Wizard from `ORACLE_HOME/oracle_common/common/bin/config.sh` directory:

   On UNIX, issue this command: `./config.sh`

   On Windows, double-click `config.exe`

6. On **Create Domain** screen, select **Create a New Domain** and provide the domain location. Click **Next**.
7. On **Templates** screen, select `Oracle Internet Directory ( Collocated ) - 14.1.2.1.0 [oid]` template. Retain all the selected dependent templates. Click **Next**.
8. On **Administrator Account** screen, provide weblogic user password and click **Next**.
9. On **Domain Mode and JDK** screen, select **Production** in **Domain Mode** field.and select the **Oracle HotSpot JDK** in the **JDK** field. . Click **Next**.

> **Note:**
>
> As of WebLogic Server 14.1.2.0.0, when you select **Production** mode, WebLogic Server automatically sets some of the security configurations of **Secured Production** to more secure values. However, there are certain security configurations (such as SSL/TLS) that require manual configuration. See Using Secured Production Mode in *Administering Security for Oracle WebLogic Server*.
>
> If you want to disable the more secure default settings, then you may select **Disable Secure Mode**. This will enable the non-SSL listen ports.
>
> If you want to retain the more secure default settings of **Secured Production** mode in general, but want to change which ports (listen ports, SSL listen ports, or administration ports) will be enabled by default in your domain, then you may:
>
> - Leave **Disable Secure Mode** unselected, and
> - Change the default port selections under **Enable or Disable Default Ports for Your Domain**
>
> For more information, see Understand How Domain Mode Affects the Default Security Configuration in *Securing a Production Environment for Oracle WebLogic Server*.

10. On **Database Configuration Type** screen, specify the database connection parameters. Change the schema owner field value from `DEV_STB` to relevant prefix — `<PREFIX_STB>` — as needed, click **Get RCU Configuration** and click **Next**.
11. On **Component Datasources** screen, click **Next**.
12. On **JDBC Test** screen, after successful connection test, click **Next**.
13. On **Advanced Configuration** screen, select **Administration Server**, **Node Manager** and **Topology**. Click **Next**.
14. On **Administration Server** screen, update **Listen Address** to a desired hostname and **Listen Port** as needed. Click **Next**.
15. On **Node Manager Type** screen, provide *Node Manager credentials* and Click **Next**.
16. On **Manager Server**, **Skip** the screen and click **Next**.
17. On **Cluster** screen, **Skip** and click **Next**.
18. On **Server Templates** screen, **Skip** and click **Next**.
19. On **Coherence Clusters** screen, **Skip** and click **Next**.
20. On **Machines** screen, Do Not change the name of the default machine name as *oidhost1*. Update the **Listen Address** to appropriate host name. The Node Manager Listen Port can be changed, if required. This port number should be the same value as the one in `nodemanager.properties` file. Add a new machine with name - *oidhost2* and update

the Listen Address to appropriate host name that points to OIDHOST2. If required, change the Listen Port to a desired port value.

Leave Node Manager type as SSL to prevent issues when connecting to Node Manager to start services.

21. On **Assign Servers to Machines** screen, select **oidhost1** and assign **AdminServer** to **oidhost1**. Click **Next**.
22. On **Virtual Targets** screen, click **Next**.
23. On **Partitions** screen, click **Next**.
24. On **Configuration Summary**, click **Create**.
25. Start **Administration Server**.
26. Start **Node Manager**.
27. From `ORACLE_HOME/oracle_common/common/bin/pack.sh` directory, execute `pack.sh` command, as shown below:

```
pack.sh -domain=<DOMAIN_HOME_LOCATION> -template=./base_domain.jar -
template_name=base_domain -managed=true
```

Configuring Oracle Internet Directory on OIDHOST2

Ensure that the OID repository is running and then follow these steps to configure the OID instance on OIDHOST2:

1. Ensure that the system, patch, kernel and other requirements are met. These are listed in Preparing to Install in *Oracle Fusion Middleware Installation Guide for Oracle Identity Management*guide.
2. Ensure that Oracle Identity Management software has been installed and upgraded on OIDHOST2 as described in Installing Oracle Fusion Middleware Components
3. On OIDHOST1, ports 3060 and 3131 were used for OID. The same ports should be used for the OID instance on OIDHOST2. Therefore, ensure that ports 3060 and 3131 are not in use by any service on OIDHOST2 by issuing these commands for the operating system you are using. If a port is not in use, no output is returned from the command.

   On Unix:

   ```
   netstat -an | grep LISTEN | grep ":3060"

   netstat -an | grep LISTEN | grep ":3131"
   ```

   On Windows:

   ```
   netstat -an | findstr "LISTEN" | findstr ":3060"

   netstat -an | findstr "LISTEN" | findstr ":3131"
   ```

4. If the port is in use (if the command returns output identifying the port), you must free the port.

   On Unix:
   Remove any entries for ports 3060 and 3131 in the `/etc/services` file and restart the services, or restart the computer. You can also check for any existing processes that is using these ports, using `netstat -anp` command.

   On Windows:
   Stop the component that is using these ports.

5. From `ORACLE_HOME/oracle_common/common` directory, create the domain using `unpack.sh` command. Use the packed domain jar file created in OIDHOST1:

   ```
   unpack.sh -template=./base_domain.jar -domain=<ORACLE_HOME>/user_projects/
   domains/base_domain
   ```

6. From `ORACLE_HOME/oracle_common/common/bin/wlst.sh` directory, Run `wlst.sh` and execute the following commands:

   ```
   connect('weblogic','<password>', 't3://<admin-host>:<admin-port>')
   oid_setup(orcladminPassword='<desired-password>', odsPassword='ODS-schema-
   password")
   oid_createInstance(instanceName='oid2', machine='oidhost2',port='oid-non-
   ssl-port',sslPort='oid-ssl-port', host='hostname-of-OIDHOST2')
   exit()
   ```

7. From the`DOMAIN_HOME/bin` directory, start Node Manager.

   ```
   ./startNodeManager.sh
   ```

8. From `DOMAIN_HOME/bin` directory, Start **oid2** instance by executing `startComponent.sh` script. Execute the script from **OIDHOST1** machine, where AdminServer is setup and not from OIDHOST2.

   - `./startComponent.sh oid2`
   - `oid2` can also be started either from OIDHOST1 or OIDHOST2 using WLST command — `nmStart()`

     ```
     nmStart(erverName='oid2', serverType='OID')
     ```

Validating Oracle Internet Directory High Availability

Use the `ldapbind` command-line tool to ensure that you can connect to each OID instance and the LDAP Virtual Server. The `ldapbind` tool enables you to determine whether you can authenticate a client to a server.

> **Note:**
>
> See the Configuring Your Environment section of *Oracle Fusion Middleware Reference for Oracle Identity Management* for a list of the environment variables you must set before using the`ldapbind` command.

For non-SSL:

```
ldapbind -h oidhost1.example.com -p 3060 -D "cn=orcladmin" -q
ldapbind -h oidhost2.example.com -p 3060 -D "cn=orcladmin" -q
ldapbind -h oid.example.com -p 3060 -D "cn=orcladmin" -q
```

> **Note:**
>
> The -q option prompts the user for a password. LDAP tools are modified to disable the options -w *password* and -P *password* when the environment variable LDAP_PASSWORD_PROMPTONLY is set to TRUE or 1. Use this feature whenever possible.

For SSL:

```
ldapbind -h oidhost1.example.com -p 3131 -D "cn=orcladmin" -q -U 1
ldapbind -h oidhost2.example.com -p 3131 -D "cn=orcladmin" -q -U 1
ldapbind -h oid.example.com -p 3131 -D "cn=orcladmin" -q -U 1
```

where -U is an optional argument used to specify the SSL authentication mode. These are the valid values for the SSL authentication mode:

- 1 = No authentication required
- 2 = One way authentication required. With this option, you must also supply a wallet location (-W "*file:/home/my_dir/my_wallet*") and wallet password (-P *wallet_password*).
- 3 = Two way authentication required. With this option, you must also supply a wallet location (-W "*file:/home/my_dir/my_wallet*") and wallet password (-P *wallet_password*).

For more information about the `ldapbind` command, see the ldapbind section in *Oracle Fusion Middleware Reference for Oracle Identity Management*.

For information about setting up SSL for OID, see Configuring Secure Sockets Layer (SSL) in the *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory* manual.

WebLogic Server Administration Console:

```
http://oidhost1.example.com:7001/console
```

Oracle Enterprise Manager Fusion Middleware Console:

```
http://oidhost1.example.com:7001/em
```

Oracle Internet Directory Failover and Expected Behavior

This section describes how to perform a failover of Oracle Internet Directory and Oracle RAC.

This section includes the following topics:

Performing Oracle Internet Directory Failover

This procedure describes the steps to be followed to perform Oracle Internet Directory failover.

The following example describes how to perform a failover to OIDHOST2 and check the status of OID service:

1. On OIDHOST1, use the following WLST command to stop the OID instance:

   ```
   shutdown(name='instance-name')
   ```

2. On OIDHOST2, check the status of OID using the load balancing router.

> **Note:**
>
> See Configuring Your Environment in *Oracle Fusion Middleware Reference for Oracle Identity Management* for a list of environment variables you must set before using the ldapbind command.

```
ldapbind -h oid.example.com -p 3060 -D "cn=orcladmin" -q
```

> **Note:**
>
> The -q option above prompts you for a password. LDAP tools are modified to disable the options -w password and -P password when the environment variable LDAP_PASSWORD_PROMPTONLY is set to TRUE or 1. Use this feature whenever possible.

Related Topics

- Managing Oracle Internet Directory Components by Using WLST Commands

Performing an Oracle RAC Failover

The `orclfailoverenabled` attribute is a configuration entry `("cn=configset,cn=oidmon,cn=subconfigsubentry")` that configures failover for Oracle Internet Directory processes. This attribute specifies the failover time in minutes before the OID Monitor will start failed processes on a surviving node. The default failover time is 5 minutes. A value of zero (0) specifies that Oracle Internet Directory processes will not fail over to another node.

To perform an Oracle RAC failover, perform the following steps:

1. Use the srvctl command to stop a database instance:

   ```
   srvctl stop instance -d db_unique_name -i inst_name_list
   ```

2. Use the srvctl command to check the status of the database:

   ```
   srvctl status database -d db_unique_name -v
   ```

3. Check the status of Oracle Internet Directory:

> **Note:**
>
> See Configuring Your Environment in *Oracle Fusion Middleware Reference for Oracle Identity Management* for a list of environment variables you must set before using the ldapbind command.

```
ldapbind -h oid_host1 -p 3060 -D "cn=orcladmin" -q
ldapbind -h oid_host2 -p 3060 -D "cn=orcladmin" -q
ldapbind -h oid.example.com -p 3060 -D "cn=orcladmin" -q
```

> **Note:**
>
> The -q option above prompts the user for a password. LDAP tools are modified to disable the options -w password and -P password when the environment variable LDAP_PASSWORD_PROMPTONLY is set to TRUE or 1. Use this feature whenever possible.

To know more about RAC failover, See Oracle Internet Directory Replication-Server Control and Failover

Troubleshooting Oracle Internet Directory High Availability

This section provides information that can help you troubleshoot OID high availability issues:

- Log files for OID are in directory:
  `DOMAIN_HOME/servers/OID/logs/InstanceName`
- The order in which log files should be examined when troubleshooting is:
  1. `oidmon-xxx.log`
  2. `oiddispd01-xxxx.log`
  3. `oidldapd01s-xxxx.log`
- This section shows some of the error messages that may be related to high availability, and their meaning:

  Error: ORA-3112, ORA-3113 errors in the log file

  Cause: one of the database node is down, OID connects again to surviving node.

  Action: See why database node went down or Oracle process got killed

  Error: Failing Over...Please stand by in the log file

  Cause: OID server received a notification from the Oracle process that one of the database node is down. OID will connect to the surviving node.

  - If the failover is successful you would see this message:

    Failover ended...resuming services.
  - If the failover was not successful, you would see these errors:
  - Tried 10 times, now quitting from failover function...
  - Bad Failover Event:
  - Forcing Failover abort as setting of DB parameters for the session failed
- If high availability event notification is enabled, you would see a message similar to the following:

```
HA Callback Event
Thread Id: 8
Event type: 0
HA Source: OCI_HA_INSTANCE
Host name: dbhost1
Database name: orcl
Instance name: orcl1
Timestamp: 14-MAY-09 03.25.24 PM -07:00
```

```
Service name: orcl.example.com
HA status: DOWN - TAF Capable
```

- If TAF is disabled, HA status will be shown as

  `DOWN`.

Action: See why database node went down.

Error: Time Difference of at least 250 sec found between node1 and node2.

Cause: There is time difference between the two nodes

Action: Synchronize the system time.

Error: Node=% did not respond for configured %d times, Failing over...

Cause: One of the OID nodes (oidmon) is not responding.

Action: See if the node is alive or OIDMON process is running.

Related Topics

- Troubleshooting Oracle Internet Directory

Additional Oracle Internet Directory High Availability Issues

This section describes issues for Oracle Internet Directory in a high availability environment.

This section describes issues for Oracle Internet Directory in a high availability environment.

See Changing the Password of the ODS Schema Used by Oracle Internet Directory

Changing the Password of the ODS Schema Used by Oracle Internet Directory

You can change the OID database schema password (that is, the password of the ODS user in the database) using the Oracle Internet Directory Database Password Utility (oidpasswd) from OIDHOST1 (Where AdminServer is installed). However, since the ODS schema password is stored in a password wallet under the DOMAIN_HOME on each host. This is propagated from OIDHOST1 to all other hosts automatically by Weblogic Domain Framework.

To change the ODS database user password, invoke the following command on one of the OID nodes:

```
oidpasswd connect=database-connection-string change_oiddb_pwd=true
```

Oracle Directory Integration Platform High Availability

This section describes how to design and deploy a high availability environment for Oracle Directory Integration Platform (ODIP).

Understanding Oracle Directory Integration Platform Component Architecture

Oracle Directory Integration Platform is a J2EE application that enables you to integrate your applications and directories, including third-party LDAP directories, with an Oracle back-end directory: Oracle Internet Directory, Oracle Unified Directory, and Oracle Directory Server Enterprise Edition.

> **Note:**
>
> Oracle Directory Integration Platform does not support Oracle Directory Server Enterprise Edition in high availability mode in this release.

See Introduction to Oracle Directory Integration Platform in *Oracle Fusion Middleware Administering Oracle Directory* for more on Oracle Directory Integration Platform architecture.

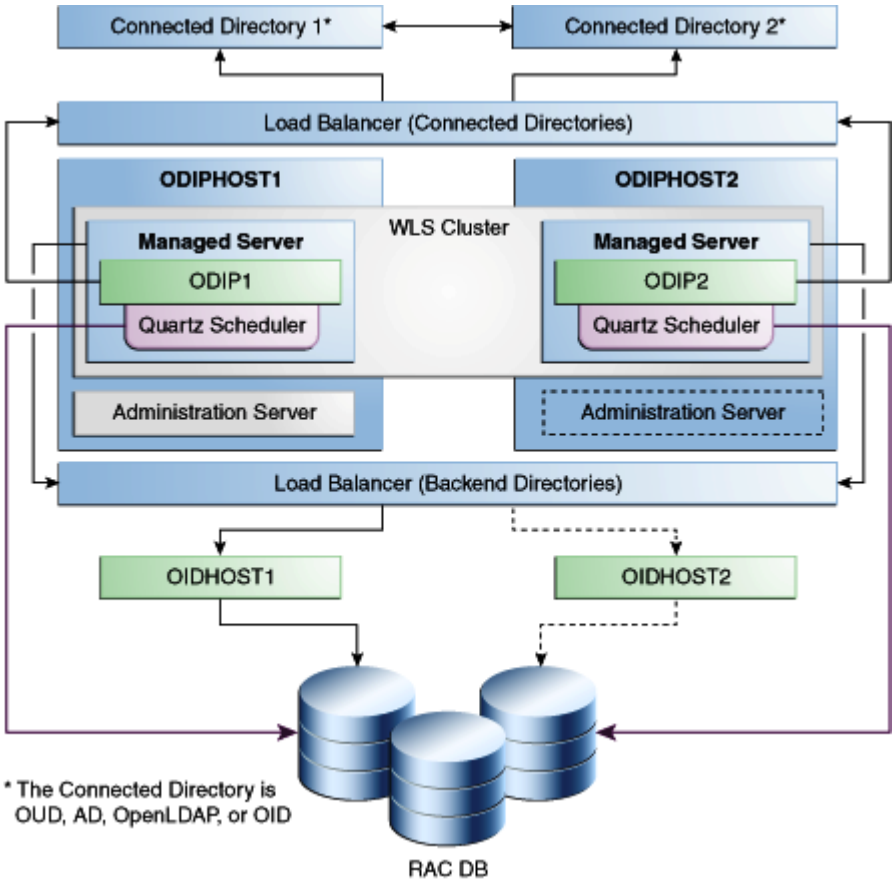
Understanding Oracle Directory Integration Platform High Availability Concepts

This section describes the Oracle Directory Integration Platform high availability concepts.

About Oracle Directory Integration Platform High Availability Architecture (OID Back-End)

Learn about the Oracle Directory Integration Platform high availability architecture with Oracle Internet Directory as the back-end directory.

Figure 7-2 Oracle Directory Integration Platform with Oracle Internet Directory (Back-End Directory) in a High Availability Architecture



In Figure 7-2 , Connected Directory 1 and Connected Directory 2 replicate information with each other. A load balancing router routes requests to the Connected Directories.

The Application Tier includes the ODIPHOST1 and ODIPHOST2 computers.

ODIP1 and ODIP2 go through the load balancer when they must communicate with the Connected Directories.

On ODIPHOST1, the following installations are performed:

- An Oracle Directory Integration Platform instance is installed (ODIP1) on the Managed Server.
- A Quartz Scheduler is installed on ODIP1 by default. It connects to the Oracle RAC database using a WebLogic multi data source. The Quartz Scheduler invokes EJBs that do the actual work; if the EJB fails, the Quartz Scheduler marks the job as failed and reschedules it to run at later time by another EJB.
- An Administration Server is installed. Under normal operations, this is the active Administration Server.

On ODIPHOST2, the following installations are performed:

- An Oracle Directory Integration Platform instance is installed (ODIP2) on the Managed Server.
- A Quartz Scheduler is installed on ODIP2 by default. Quartz Scheduler connects to the Oracle RAC database using a WebLogic multi data source.
- An Administration Server is installed. Under normal operations, this is the passive Administration Server instance. You make this Administration Server active if the Administration Server on ODIPHOST1 becomes unavailable.

The Oracle Directory Integration Platform instances on the ODIPHOST1 and ODIPHOST2 Managed Servers are configured as a cluster.

A load balancer is set up for the back-end directories OIDHOST1 and OIDHOST2. The load balancer routes requests to either OIDHOST1 or OIDHOST2.

> **Note:**
>
> When you use a RAC database, multi data source is used with Oracle Directory Integration Platform to protect the instances from RAC failure.

About Starting and Stopping the Cluster

By default, the WebLogic Server starts, stops, and monitors the applications and Oracle Directory Integration Platform leverages the high availability features of the underlying clusters. If there is a hardware or other failure, session state is available to other cluster nodes that can resume the work of the failed node.

Node Manager monitors the WebLogic servers. If failure occurs, Node Manager restarts the WebLogic Server.

See Configuring Java Node Manager in *Administering Node Manager for Oracle WebLogic Server*.

Cluster-Wide Configuration Changes (OID)

When you deploy Oracle Internet Directory in a high availability configuration, all Oracle Internet Directory instances in the cluster share the same database. Any changes made to Oracle Directory Integration Platform on one Oracle Internet Directory node automatically propagate to all the Oracle Internet Directory instances in the cluster.

Directory Synchronization Profiles

Changes that you make to directory integration profiles on one Oracle Internet Directory node do not replicate automatically to other Oracle Internet Directory nodes in a default multimaster Oracle Internet Directory replication environment. You must copy changes from the primary node to the secondary nodes manually and do so on a periodic basis. By doing this, a directory synchronization profile can run on a secondary node if a problem occurs on the primary node.

Oracle Directory Integration Platform uses the parameter `orcllastappliedchangenumber`. The value assigned to the `lastchangenumber` attribute in a directory synchronization profile depends on the directory server on which Oracle Directory Integration Platform is running. In an active-active Oracle Directory Integration Platform configuration, you must manually update the `lastchangenumber` attribute in all instances.

To synchronize directory provisioning profiles between the primary Oracle Internet Directory node and secondary nodes:

1. On the primary node, use the `ldifwrite` command to create an LDIF dump of the entries from this container:

   ```
   cn=subscriber profiles,cn=changelog subscriber,cn=oracle internet directory
   ```

2. Copy the LDIF dump to the secondary node.
3. Use the `ldapadd` command to add the profiles on the secondary node.

After you copy an export profile to a target node, you must update the `lastchangenumber` attribute with the target node value. To update the value:

1. Disable the synchronization profile.
2. Get the value of the `lastchangenumber` attribute on the target node using the `ldapsearch` command.
3. Use `ldapsearch` to get the LDIF dump of the profile entry.
4. Use `ldapadd` to add the profile to the other Managed Server instance.
5. Go to the Oracle Directory Integration Platform Admin console and select the profile. Select **Edit**. Select the **Advanced** tab then select **Edit and Persist**. Enter the value of the `lastchangenumber` attribute. Save the profile.
6. Enable the synchronization profile.

Directory Provisioning Profiles

In a default multimaster Oracle Internet Directory replication environment, Oracle Directory Integration Platform is installed in the same location as the primary Oracle Internet Directory. The information and steps in this topic applies only when multimaster replication is set up.

If the primary node fails, event propagation stops for all profiles located on the node. Although the events are queued and not lost while the primary node is stopped, the events do not propagate to any applications that expect them. To ensure that events continue to propagate

even when the primary node is down for the Version 1.0 and 2.0 profiles, the directory provisioning profiles must be copied to other secondary nodes.

However, copy directory provisioning profiles from the primary node to any secondary nodes immediately after an application is installed and before any user changes are made in Oracle Internet Directory.

To synchronize directory provisioning profiles between a primary node and any secondary nodes:

1. On the primary node, use the ldifwrite command to create an LDIF dump of the entries from this container:

   ```
   cn=provisioning profiles,cn=changelog subscriber,cn=oracle internet directory
   ```

2. Copy the LDIF dump to the secondary node.
3. Use the `ldapadd` command to add the profiles on the secondary node.

About Oracle Directory Integration Platform High Availability Architecture (OUD Back-End)

This section describes the Oracle Directory Integration Platform high availability architecture with Oracle Unified Directory (OUD) as the back-end directory.

Figure 7-3 Oracle Directory Integration Platform with Oracle Unified Directory (Back-End Directory) in a High Availability Architecture

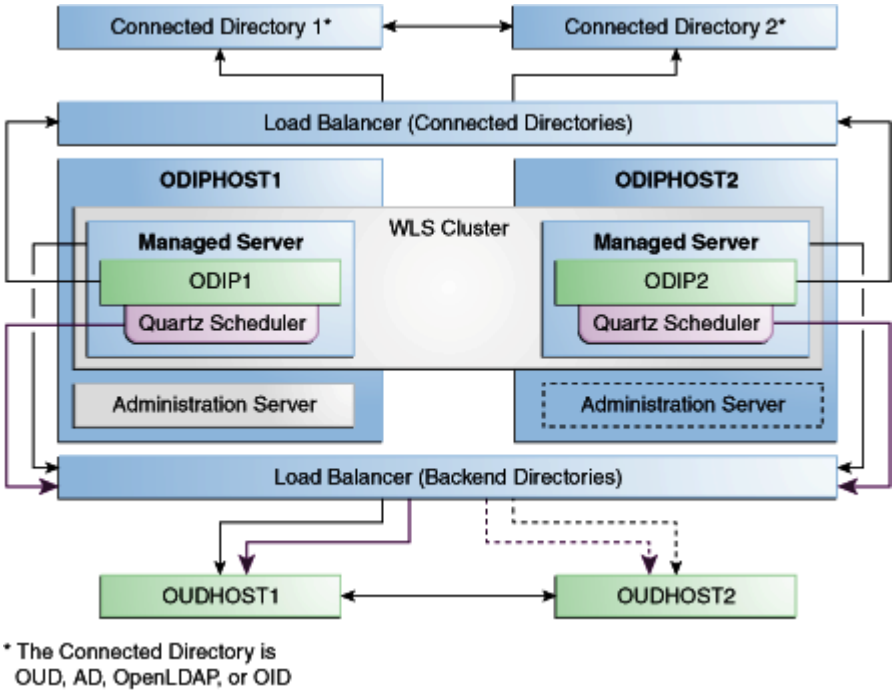


Figure 7-3, Connected Directory 1 and Connected Directory 2 replicate information with each other. A load balancing router routes requests to the Connected Directories.

The Application Tier includes the `ODIPHOST1` and `ODIPHOST2` computers.

On `ODIPHOST1`, the following installations are performed:

- An Oracle Directory Integration Platform instance is installed (ODIP1) on the Managed Server. ODIP1 goes through the load balancer for connected directories when it must connect to them.
- The Quartz Scheduler is installed. It goes through the load balancer for the back-end directories.
- An Administration Server is installed. Under normal operations, this is the active Administration Server.

On ODIPHOST2, the following installations are performed:

- An ODIP instance is installed (ODIP2) on the Managed Server. ODIP2 goes through the load balancer for connected directories when it must connect to them.
- The Quartz Scheduler is installed. It goes through the load balancer for backend directories.
- An Administration Server is installed. Under normal operations, this is the passive Administration Server instance. You make this Administration Server active if the Administration Server on ODIPHOST1 becomes unavailable.

The Oracle Directory Integration Platform instances on the ODIPHOST1 and ODIPHOST2 Managed Servers are configured as a cluster.

A load balancer is set up for the back-end directories OUDHOST1 and OUDHOST2. The load balancer routes requests to either OUDHOST1 or OUDHOST2.

Cluster-Wide Configuration Changes (OUD)

Oracle Unified Directory supports cluster-wide configuration changes. All Oracle Unified Directory instances that are part of the same replication topology share the same content. Any changes made to Oracle Directory Integration Platform on one Oracle Unified Directory node automatically propagate to all Oracle Unified Directory instances in the replication topology.

Protection from Failures and Expected Behavior

This section describes protection from different types of failure in an Oracle Directory Integration Platform active-active cluster

About Process Failure

In a high availability environment, you deploy the Oracle Directory Integration Platform application to a cluster that comprises at least two Oracle WebLogic instances.

By default, the Oracle Directory Integration Platform application leverages high availability features of the underlying WebLogic clusters. When you deploy Oracle Directory Integration Platform, the Quartz scheduler starts with a clustering option. Depending on the load on the node, the scheduler then runs the job on any available nodes in the cluster. If hardware or other failures occur on one or more nodes, the Quartz scheduler runs the jobs on available nodes.

Also, Node Manager monitors WebLogic servers. In case of failure, Node Manager restarts the WebLogic server.

Within the Oracle Directory Integration Platform application, the Quartz Scheduler invokes the Provisioning or Synchronization EJBs that do the actual work. As soon as the Quartz scheduler invokes an EJB, it tags that EJB as running the job. If the EJB fails, the Quartz scheduler marks the job as failed and reschedules it to run later by another EJB.

About Updating the Oracle Directory Integration Platform Server Configuration

If the back-end server is not accessed or cannot be accessed through a load balancer, Oracle Directory Integration Platform failover is not transparent.

This scenario requires manual intervention because the information to connect to the back-end directory is local to each Oracle Directory Integration Platform instance.

You must run the `manageDIPServerConfig` utility to update the Oracle back-end directory (Oracle Internet Directory and Oracle Unified Directory) `host` and `port` parameters for all of the Oracle Directory Integration Platform instances.

See manageDIPServerConfig Utility in *Oracle Fusion Middleware Administering Oracle Directory Integration Platform*.

About External Dependency Failure

Oracle Directory Integration Platform requires the back-end repository, Oracle Internet Directory, Oracle Unified Directory, Credential Store Framework, and the WebLogic Managed Server to be available during startup.

It fails to start if any one of these elements are unavailable.

Configuring Oracle Directory Integration Platform for High Availability

You can use Oracle Internet Directory or Oracle Unified Directory as the as the back-end directory to configure Oracle Directory Integration Platform high availability.

Configuring High Availability for an Oracle Internet Directory Back-End Server

Use the steps in the following order to configure Oracle Internet Directory (back-end directory) for Oracle Directory Integration Platform high availablity.

- Before You Configure Oracle Directory Integration High Availability (OID)
- Configuring Oracle Directory Integration Platform on ODIPHOST1 (OID)
- Configuring Oracle Directory Integration Platform for Oracle Internet Directory (OIDHOST1)
- Configuring Oracle Directory Integration Platform on ODIPHOST2 (OID)

Before You Configure Oracle Directory Integration High Availability (OID)

Complete the following before you configure Oracle Directory Integration Platform high availability with Oracle Internet Directory as the back-end directory:

- Ensure that Oracle Internet Directory is configured for high availability, as described in Oracle Internet Directory High Availability Configuration Steps.
- Oracle WebLogic Server and Oracle Directory Integration Platform is installed across all nodes (`ODIPHOST1` and `ODIPHOST2`).

Configuring Oracle Directory Integration Platform on ODIPHOST1 (OID)

To configure Oracle Directory Integration Platform on `ODIPHOST1`:

1. Start the Configuration Wizard by running the `ORACLE_HOME/oracle_common/common/bin/config.sh` script (on UNIX) or `ORACLE_HOME\oracle_common\common\bin\config.cmd` (on Windows).

   The **Configuration Type** screen is displayed.

2. Select **Update an existing domain**, and click **Next**.

   The **Templates** screen is displayed.

3. On the **Templates** screen, select **Update Domain Using Product Templates** and then select **Oracle Directory Integration Platform - 14.1.2.1.0 [dip]** domain configuration option.

   > **Note:**
   >
   > When you select the **Oracle Directory Integration Platform - 14.1.2.1.0 [dip]** option, **Oracle Enterprise Manager 14.1.2.0.0 [em]** is automatically selected.

   Click **Next**.

   The **JDBC Data Sources** screen is displayed.

4. Make changes if required and then click **Next**

   The **JDBC Data Sources Test** screen is displayed.

5. Select the data sources to test, and click **Test Selected Connections**.

   Click **Next**.

   The **Database Configuration Type** screen is displayed.

6. Make changes if required and then click **Get RCU Configuration** to retrieve the schema information. After successfully retrieving the schema information, click **Next** to continue.

   The **JDBC Component Schema** screen is displayed.

7. Verify that the values populated are correct for all schemas, and Click **Next**.

   > **Note:**
   >
   > To convert one or more of the schemas to Oracle RAC multi-data source schemas, select the check boxes next to the name of those schemas, and select the **Convert to RAC multi data source** option. Click Next when done. When you click Next, the **Oracle RAC Multi Data Source Component Schema** screen appears.
   >
   > See Oracle RAC Multi Data Source Component Schema in *Oracle Fusion Middleware Creating WebLogic Domains Using the Configuration Wizard*.

   The **JDBC Component Schema Test** screen is displayed.

8. You can select the component schema to test, and click **Test Selected Connections**. Wait for one or more connection tests to complete. If you do not want to test connections, deselect all data sources.

> **Note:**
>
> In order to test connections, the database to which you are trying to connect must be running.

Click **Next**.

The **Advanced Configuration** screen is displayed.

9. Select **Managed Servers, Clusters, and Coherence** option. Click **Next**.

   The **Managed Servers** screen is displayed.

10. Click **Add**, and create one Managed Servers each for `ODIPHOST1` and `ODIPHOST2`.

Table 7-6 Managed Server on ODIPHOST1

Name	Listen Address	Listen Port
`wls_ods1`	`odipHost1.example.com`	7005

Table 7-7 Managed Server on ODIPHOST2

Name	Listen Address	Listen Port
`wls_ods2`	`odipHost2.example.com`	7005

Click **Next**.

The **Clusters** screen is displayed.

11. Click **Add** and enter `odip_cluster` in the **Cluster Name** field to configure cluster for the Managed Servers on `ODIPHOST1` and `ODIPHOST2`.

    Click **Next**.

    The **Server Templates** screen is displayed.

12. Click **Next** and **Dynamic Servers** screen is displayed.

    Click **Next**.

    The **Assign Servers to Clusters** screen is displayed.

13. Use the Assign Servers to Clusters screen to assign the `wls_ods1` and `wls_ods2` Managed Servers to the `odip_cluster` cluster. Only Managed Servers appear in the Server list box. The Administration Server is not listed because it cannot be assigned to a cluster.

    Select the name of the Managed Server in the **Servers** list box and click the right arrow. The name of the Managed Server is removed from the **Servers** list box and added below the name of the target cluster in the **Clusters** list box.

    The name of the Managed Server is removed from the Servers list box and added below the name of the target cluster in the Clusters list box.

    Click **Next** and continue clicking **Next** till the **Machines** screen is displayed.

14. Click the **Machine** (for Windows) or **Unix Machine** tab (for UNIX) and then click **Add** to add the following machines:

Table 7-8 Machines

Name	Node Manager Listen Address	Node Manager Listen Port
odip_1	odipHost1.example.com	5556
odip_2	odipHost2.example.com	5556

Click **Next**.

The **Assign Servers to Machines** screen is displayed.

15. Use the **Assign Servers to Machines** to assign the WebLogic Server instances to each of the machines.
    a. In the **Machine** list box, select the odip_1 machine.
    b. Select the wls_ods1 instance in the **Server** list box and click the right arrow.

       The name of the wls_ods1 instance is removed from the **Server** list box and added, below the name of the target machine, in the **Machine** list box.
    c. Repeat above steps to assign odip_2 machine to the wls_ods2 Managed Server.

    Select the name of the Managed Server in the **Servers** list box and click the right arrow. The name of the Managed Server is removed from the **Servers** list box and added below the name of the target cluster in the **Clusters** list box.

    The name of the Managed Server is removed from the Servers list box and added below the name of the target cluster in the Clusters list box.

    Click **Next** and continue clicking **Next** till the **Configuration Summary** screen is displayed.
16. Review each item on the **Configuration Summary** 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**.

    A new WebLogic domain (for example: *base_domain*) is created to support Oracle Directory Integration Platform and Fusion Middleware Control in the <ORACLE_HOME>\user_projects\domains directory (on Windows). On UNIX, the domain is created in the <ORACLE_HOME>/user_projects/domains directory.

Configuring Oracle Directory Integration Platform for Oracle Internet Directory (OIDHOST1)

You must configure Oracle Directory Integration Platform for Oracle Internet Directory on OIDHOST1 instance.

Complete the following steps:

1. Run the dipConfigurator command to configure Oracle Directory Integration Platform (ODIPHOST1) for OIDHOST1. For more information, see Configuring Oracle Directory Integration Platform for Oracle Internet Directory in *Oracle Fusion Middleware Administering Oracle Directory Integration Platform*.

> **Note:**
>
> - If you are using a RAC database, then Oracle recommends that you specify the URL for the RAC database in the `dbconfigfile` file for `dipConfigurator` properties.
> - If the cipher suites configured for Oracle Internet Directory are not available or recognized in Oracle Directory Integration Platform then you must add those suites into Oracle Directory Integration Platform using the Oracle Fusion Middleware System MBean Browser. See Adding Cipher Suites Configured for Oracle Internet Directory into Oracle Directory Integration Platformin *Oracle Fusion Middleware Administering Oracle Directory Integration Platform*.

2. Run the `manageDIPServerConfig` command to tune the cluster:

```
./manageDIPServerConfig set -host ODIPHOST1.example.com -port 7005 -wlsuser weblogic
-attribute ClusterCheckInInterval -value 30000

./manageDIPServerConfig set -host ODIPHOST1 -port 7005 -wlsuser weblogic -attribute
RefreshInterval -value 120
```

3. Run the `manageDIPServerConfig` command for reconfiguring Oracle Directory Integration Platform to use the TCP load balancer.

   `LB_HOST` is the load balancer IP address you must configure to redirect to one of the back-end instances.

```
./manageDIPServerConfig set -host ODIPHOST1 -port 7005 -wlsuser weblogic -attribute
BackendHostPort -value LB_HOST:LB_PORT
```

Configuring Oracle Directory Integration Platform on ODIPHOST2 (OID)

You must configure the Oracle Directory Integration Platform on `ODIPHOST2` for the Oracle Internet Directory back-end directory:

1. Run the following pack command on `ODIPHOST1` to create a template pack:

```
cd MW_HOME/oracle_common/common/bin
./pack.sh -managed=true -domain=MW_HOME/user_projects/domains/domainName -
template=dipdomain.jar -managed=true -template_name="dipdomain"
```

2. Copy the template file created in the previous step from `ODIPHOST1` to `ODIPHOST2`. For example, on a UNIX platform:

```
scp dipdomain.jar user@ODIPHOST2:MW_HOME/oracle_common/common/bin
```

3. Perform the following on `ODIPHOST2`:

   a. Run the `unpack` command to unpack the propagated template:

```
cd MW_HOME/oracle_common/common/bin
./unpack.sh -domain=MW_HOME/user_projects/domains/domains/domainName -
template=dipdomain.jar -overwrite_domain=true
```

   b. Start and stop the `wls_ods2` Managed Server:

```
MW_HOME/user_projects/domains/domainName/bin/startManagedWebLogic.sh wls_ods2
http://ODIPHOST1:ODIPHOST1ADMINPORT

MW_HOME/user_projects/domains/domainName/bin/stopManagedWebLogic.sh wls_ods2
http://ODIPHOST1:ODIPHOST1ADMINPORT
```

c. Overwrite the `dip-config.xml` file in `wls_ods2` with the `dip-config.xml` in `wls_ods1`:

```
cp MW_HOME/user_projects/domains/DOMAIN_NAME/config/fmwconfig/servers/wls_ods1/
applications/DIP_14.1.2.1.0/configuration/dip-config.xml
 MW_HOME/user_projects/domains/DOMAIN_NAME/config/fmwconfig/servers/wls_ods2/
applications/DIP_14.1.2.1.0/configuration/dip-config.xml
```

d. Start the Node Manager, by running the `startNodeManager.cmd` (Windows) or `startNodeManager.sh` (UNIX) command.

```
MW_HOME/user_projects/domains/DOMAIN_NAME/bin/startNodeManager.sh
```

e. Start the `wls_ods2` Managed Server:

```
MW_HOME/user_projects/domains/DOMAIN_NAME/bin/startManagedWebLogic.sh wls_ods2
http://ODIPHOST1:ODIPHOST1ADMINPORT
```

Configuring High Availability for an Oracle Unified Directory Back-End Server

Use the steps in the following order to configure Oracle Unified Directory (back-end directory) for Oracle Directory Integration Platform high availability.

Before You Configure Oracle Directory Integration High Availability (OUD)

Complete the following before you configure Oracle Directory Integration Platform high availability with Oracle Unified Directory as the back-end directory:

- Ensure that you install Oracle Unified Directory, see Installing the Oracle Unified Directory Software in *Oracle Fusion Middleware Installing Oracle Unified Directory*.

  When you set up an Oracle Unified Directory server instance using either the graphical user interface (GUI) or the command-line interface (CLI), ensure that you select the **Enable for DIP** option to enable the server instance for Oracle Directory Integration Platform.
- Ensure that Oracle Unified Directory is configured for high availability. See Understanding Oracle Unified Directory High Availability Deployments in *Oracle Fusion Middleware Administering Oracle Unified Directory*.
- Ensure that you have created the Oracle Unified Directory Suffixes for Oracle Directory Integration Platform. See Creating Oracle Unified Directory Suffixes in *Oracle Fusion Middleware Administering Oracle Directory Integration Platform*.
- Ensure that the change log is enabled. See Enabling External Change Login *Oracle Fusion Middleware Administering Oracle Directory Integration Platform*.
- Oracle WebLogic Server and Oracle Directory Integration Platform is installed across all nodes (`ODIPHOST1` and `ODIPHOST2`).

Configuring Oracle Directory Integration Platform on ODIPHOST1 (OUD)

To configure Oracle Directory Integration Platform on `ODIPHOST1` for Oracle Unified Directory as the back-end directory:

1. Start the Configuration Wizard by running the `<MW_HOME>/oracle_common/common/bin/config.sh` script (on UNIX) or `<MW_HOME>\oracle_common\common\bin\config.cmd` (on Windows).

   The **Configuration Type** screen is displayed.

2. On the **Configuration Type** screen, select **Create a new domain** and enter the full path for the domain or use the **Browse** button to navigate to the directory in which your domains are located. Click **Next**.

   The **Templates** screen is displayed.

3. On the **Templates** screen, make sure **Create Domain Using Product Templates** is selected, and then select **Oracle Directory Integration Platform - 14.1.2.1.0 [dip]**.

   > **Note:**
   >
   > When you select **Oracle Directory Integration Platform - 14.1.2.1.0 [dip]** option, the following components are automatically selected:
   >
   > - **Oracle Enterprise Manager 14.1.2.0.0 [em]**
   > - **Oracle JRF - 14.1.2.0.0 [oracle_common]**
   > - **Weblogic Coherence Cluster Extension 14.1.2.0.0 [wlserver]**

   Click **Next**.

   Click The **Application Location** screen is displayed.

4. Click **Browse** and specify the full path to the directory in which you want to store the applications that are associated with the domain.

   Click **Next**.

   The **Administrator Account** screen is displayed.

5. Specify the user name and password for the default WebLogic Administrator account for the domain.

   The password must be at least eight characters and must contain at least one number or special character. Confirm the password and click **Next**.

   Make a note of these details as you will need them to start or restart the WebLogic domain in the following procedure.

   The **Domain Mode and JDK** screen is displayed.

6. Specify the domain mode and Java Development Kit (JDK).

   a. Select **Production** in the Domain Mode field.

> **Note:**
>
> As of WebLogic Server 14.1.2.0.0, when you select **Production** mode, WebLogic Server automatically sets some of the security configurations of **Secured Production** to more secure values. However, there are certain security configurations (such as SSL/TLS) that require manual configuration. See Using Secured Production Mode in *Administering Security for Oracle WebLogic Server*.
>
> If you want to disable the more secure default settings, then you may select **Disable Secure Mode**. This will enable the non-SSL listen ports.
>
> If you want to retain the more secure default settings of **Secured Production** mode in general, but want to change which ports (listen ports, SSL listen ports, or administration ports) will be enabled by default in your domain, then you may:
>
> - Leave **Disable Secure Mode** unselected, and
> - Change the default port selections under **Enable or Disable Default Ports for Your Domain**
>
> For more information, see Understand How Domain Mode Affects the Default Security Configuration in *Securing a Production Environment for Oracle WebLogic Server*.

b. Accept **Oracle Hotspot** as a default JDK location.

c. Click **Next**.

The **Database Configuration Type** screen is displayed.

7. Select **RCU Data**. This option instructs the Configuration Wizard to connect to the database's Service Table (STB) schema to automatically retrieve schema information for schemas needed to configure the domain.

> **Note:**
>
> Ensure that you have created the database schemas required for Oracle Internet Directory. See Creating the Database Schemas in *Oracle Fusion Middleware Installing and Configuring Oracle Internet Directory*.

After selecting RCU Data:

a. Enter the name of the server hosting the database in the **Host Name** field.

b. Enter the database DBMS name, or service name if you selected a service type driver in the **DBMS/Service** field.

c. Enter the port number on which the database listens.

d. Enter the username and password for connecting to the database's Service Table schema.

e. Click **Get RCU Configuration** to retrieve the schema information. After successfully retrieving the schema information, click **Next** to continue.

The **JDBC Component Schema** screen is displayed.

8. Verify that the values populated are correct for all schemas, and Click **Next**.

> **Note:**
>
> To convert one or more of the schemas to Oracle RAC multi-data source schemas, select the check boxes next to the name of those schemas, and select the **Convert to RAC multi data source** option. Click Next when done. When you click Next, the **Oracle RAC Multi Data Source Component Schema** screen appears.
>
> See Oracle RAC Multi Data Source Component Schema in *Oracle Fusion Middleware Creating WebLogic Domains Using the Configuration Wizard*.

The **JDBC Component Schema Test** screen is displayed.

9. Click **Test Selected Connection** to test datasource connections that you just configured.

   A green check mark in the Status column indicates a successful test. If you encounter issues, see the error message in the Connection Result Log section of the screen, fix the problem, then test the connection again.
   The **Advanced Configuration** screen is displayed.

10. To complete domain configuration, select the following options:

    - **Administration Server**: Required to properly configure the Administration Server's listen address.
    - **Node Manager**: Required to configure Node Manager.
    - **Topology**: Required to configure the Managed Servers and cluster, and for configuring the machine and targeting Managed Servers to the machine.

    Click **Next**.

    The **Administration Server** screen is displayed.

11. Accept the default settings or change the Administration Server settings.

    Click **Next**.

    The **Node Manager** screen is displayed.

12. Use the Node Manager screen to select the Node Manager configurations that are applicable for the domain and click **Next**.

    The **Managed Servers** screen is displayed.

13. Click **Add**, and create one Managed Servers each for `ODIPHOST1` and `ODIPHOST2`.

Table 7-9 Managed Servers on ODIPHOST1

Name	Listen Address	Listen Port
`wls_ods1`	`odipHost1.example.com`	7005

Table 7-10 Managed Servers on ODIPHOST2

Name	Listen Address	Listen Port
`wls_ods2`	`odipHost2.example.com`	7005

Click **Next**.

The **Clusters** screen is displayed.

14. Click **Add** and enter `odip_cluster` in the **Cluster Name** field to configure cluster for the Managed Servers on `ODIPHOST1` and `ODIPHOST2`.

    Click **Next**.

    The **Server Templates** screen is displayed.

15. Click **Next** and **Dynamic Servers** screen is displayed.

    Click **Next**.

    The **Assign Servers to Clusters** screen is displayed.

16. Use the Assign Servers to Clusters screen to assign the `wls_ods1` and `wls_ods2` Managed Servers to the `odip_cluster` cluster. Only Managed Servers appear in the Server list box. The Administration Server is not listed because it cannot be assigned to a cluster.

    Select the name of the Managed Server in the **Servers** list box and click the right arrow. The name of the Managed Server is removed from the **Servers** list box and added below the name of the target cluster in the **Clusters** list box.

    The name of the Managed Server is removed from the Servers list box and added below the name of the target cluster in the Clusters list box.

    Click **Next** and continue clicking **Next** till the **Machines** screen is displayed.

17. Click the **Machine** or **Unix Machine** tab and then click **Add** to add the following machines:

Table 7-11 Machines

Name	Node Manager Listen Address	Node Manager Listen Port
`odip_1`	`odipHost1.example.com`	`5556`
`odip_2`	`odipHost2.example.com`	`5556`

    Click **Next**.

    The **Assign Servers to Machines** screen is displayed.

18. Use the **Assign Servers to Machines** to assign the WebLogic Server instances to each of the machines.

    a. In the **Machine** list box, select the `odip_1` machine.

    b. Select the `wls_ods1` instance in the **Server** list box and click the right arrow.

       The name of the `wls_ods1` instance is removed from the **Server** list box and added, below the name of the target machine, in the **Machine** list box.

    c. Repeat above steps to assign `odip_2` machine to the `wls_ods2` Managed Server.

    Select the name of the Managed Server in the **Servers** list box and click the right arrow. The name of the Managed Server is removed from the **Servers** list box and added below the name of the target cluster in the **Clusters** list box.

    The name of the Managed Server is removed from the Servers list box and added below the name of the target cluster in the Clusters list box.

    Click **Next** and continue clicking **Next** till the **Configuration Summary** screen is displayed.

19. Review each item on the **Configuration Summary** 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**.

A new WebLogic domain (for example: *base_domain*) is created to support Oracle Directory Integration Platform and Fusion Middleware Control in the `<MW_HOME>\user_projects\domains` directory (on Windows). On UNIX, the domain is created in the `<MW_HOME>/user_projects/domains` directory.

Configuring Oracle Directory Integration Platform for Oracle Unified Directory (OUDHOST1)

You must configure Oracle Directory Integration Platform for Oracle Unified Directory on `OIDHOST1` instance.

Complete the following steps:

1. Run the `dipConfigurator` command to configure Oracle Directory Integration Platform (`ODIPHOST1`) for `OUDHOST1`. For more information, see Configuring Oracle Directory Integration Platform for Oracle Unified Directory in *Oracle Fusion Middleware Administering Oracle Directory Integration Platform*.
2. Run the `manageDIPServerConfig` command to tune the cluster:

   ```
   ./manageDIPServerConfig set -host ODIPHOST1.example.com -port 7005 -wlsuser weblogic
   -attribute ClusterCheckInInterval -value 30000

   ./manageDIPServerConfig set -host ODIPHOST1 -port 7005 -wlsuser weblogic -attribute
   RefreshInterval -value 120
   ```

3. Run the `manageDIPServerConfig` command for reconfiguring Oracle Directory Integration Platform to use the TCP load balancer.

   `LB_HOST` is the load balancer IP address you must configure to redirect to one of the back-end instances.

   ```
   ./manageDIPServerConfig set -host ODIPHOST1 -port 7005 -wlsuser weblogic -attribute
   BackendHostPort -value LB_HOST:LB_PORT
   ```

Configuring Oracle Directory Integration Platform on ODIPHOST2 (OUD)

You must configure the Oracle Directory Integration Platform on `ODIPHOST2` for the Oracle Unified Directory back-end directory:

1. Run the following pack command on `ODIPHOST1` to create a template pack:

   ```
   cd MW_HOME/oracle_common/common/bin
   ./pack.sh -managed=true -domain=MW_HOME/user_projects/domains/domainName -
   template=dipdomain.jar -managed=true -template_name="dipdomain"
   ```

2. Copy the template file created in the previous step from `ODIPHOST1` to `ODIPHOST2`. For example, on a UNIX platform:

   ```
   scp dipdomain.jar user@ODIPHOST2:MW_HOME/oracle_common/common/bin
   ```

3. Perform the following on `ODIPHOST2`:
   a. Run the `unpack` command to unpack the propagated template:

      ```
      cd MW_HOME/oracle_common/common/bin
      ./unpack.sh -domain=MW_HOME/user_projects/domains/domains/domainName -
      template=dipdomain.jar -overwrite_domain=true
      ```

   b. Start and stop the `wls_ods2` Managed Server:

      ```
      MW_HOME/user_projects/domains/domainName/bin/startManagedWebLogic.sh wls_ods2
      http://ODIPHOST1:ODIPHOST1ADMINPORT
      ```

```
MW_HOME/user_projects/domains/domainName/bin/stopManagedWebLogic.sh wls_ods2
http://ODIPHOST1:ODIPHOST1ADMINPORT
```

c. Overwrite the `dip-config.xml` file in `wls_ods2` with the `dip-config.xml` in `wls_ods1`:

```
cp MW_HOME/user_projects/domains/DOMAIN_NAME/config/fmwconfig/servers/wls_ods1/
applications/DIP_14.1.2.1.0/configuration/dip-config.xml
 MW_HOME/user_projects/domains/DOMAIN_NAME/config/fmwconfig/servers/wls_ods2/
applications/DIP_14.1.2.1.0/configuration/dip-config.xml
```

d. Start the Node Manager, by running the `startNodeManager.cmd` (Windows) or `startNodeManager.sh` (UNIX) command.

```
MW_HOME/user_projects/domains/DOMAIN_NAME/bin/startNodeManager.sh
```

e. Start the `wls_ods2` Managed Server:

```
MW_HOME/user_projects/domains/DOMAIN_NAME/bin/startManagedWebLogic.sh wls_ods2
http://ODIPHOST1:ODIPHOST1ADMINPORT
```

About Retrieving Changes from Connected Directories

Oracle Directory Integration Platform uses readers to retrieve changes from connected directories. However, there are some connectors that you cannot use for load-balanced directories. This section describes how Oracle Directory Integration Platform supports the use of several instances of a connected directory for import profiles.

Failing Over Oracle Directory Server Enterprise Edition Manually

Oracle Directory Integration Platform does not support transparent failover from one Oracle Directory Server Enterprise Edition (ODSEE) Managed Server (WLS_ODSEE1) to another ODSEE server (WLS_ODSEE2). Even if you replicate ODSEE Managed Server instances, the change numbers may not be identical on both ODSEE Managed Servers for the same update. If Oracle Directory Integration Platform fails over transparently from WLS_ODSEE1 to WLS_ODSEE2, ODIP may replay changes or miss changes each time it switches.

Oracle Unified Directory

When you use Oracle Unified Directory with Iplanet Reader and Iplanet Writer, Oracle Unified Directory does not support transparent failover from one Oracle Unified Directory instance to another because, as with ODSEE Server, change numbers may not be synchronized. However, you can configure your profile to use an Oracle Unified Directory connector that does support it.

To configure your profile, you must set the reader to `oracle.ldap.odip.gsi.OudCookieReader`. You must configure this attribute at creation time; you cannot configure it for existing profiles.

1. Go to the directory `ORACLE_HOME/ldap/odi/conf` and edit the file `iplanetimp.cfg.master`
2. Replace the line `Reader: oracle.ldap.odip.gsi.IPlanetReader` with the line `oracle.ldap.odip.gsi.OudCookieReader`

To failover transparently from one Oracle Unified Directory instance to another, the reader uses the External Change Log cookie that Oracle Unified Directory provides. The last applied change number contains a cookie but no longer contains a change number.

See Using the External Change Log in *Oracle Fusion Middleware Administering Oracle Unified Directory* for more information on Oracle Unified Directory external change log cookies.

Novell eDirectory

Because the Oracle Directory Integration Platform reader for Novell eDirectory is based on timestamps, clocks on all instances must be synchronized.

OpenLDAP

Because Oracle Directory Integration Platform reader for OpenLDAP is based on timestamps, clocks on all instances must be synchronized.

IBM Tivoli Directory Server

Oracle does not support IBM Tivoli by means of the load balancer.

Oracle Internet Directory

If you configure Oracle Internet Directory replication so that change numbers are identical on all Oracle Internet Directory instances that you target, the Oracle Internet Directory instances can failover transparently. If you do not set up this configuration, transparent failover is not supported.

Understanding Oracle Directory Integration Platform Failover and Expected Behavior

In a high availability environment, you deploy the Oracle Directory Integration Platform application on a WebLogic Server cluster that comprises at least two WebLogic instances.

By default, the Oracle Directory Integration Platform application leverages high availability features of the underlying WebLogic 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.

In addition, in a high availability environment, Node Manager is configured to monitor the WebLogic servers. In case of failure, Node Manager restarts the WebLogic Server.

If an instance of Oracle Internet Directory fails, the load balancer redirects to the surviving instance of Oracle Internet Directory and the Oracle RAC database. If Oracle Unified Directory fails, the load balancer redirects to the surviving instance of Oracle Unified Directory.

In case of a database instance failure, the surviving Oracle RAC node takes over any remaining processes. There may be innocuous errors in the Managed Servers logs during an Oracle RAC failover; see Troubleshooting Oracle Directory Integration Platform High Availability.

Troubleshooting Oracle Directory Integration Platform High Availability

This section describes how to manage issues involving Oracle Directory Integration Platform high availability.

Managed Server Log File Exception May Occur During an Oracle RAC Failover

During an Oracle RAC failover, exceptions similar to the ones below are seen in the Managed Server log files running the Oracle Directory Integration Platform application. These errors are thrown when the multi data sources configured on the WebLogic Server platform try to verify the health of the Oracle RAC database instances during failover. These are innocuous errors that you can ignore. The Oracle Directory Integration Platform application recovers and begins to operate normally after a lag of one or two minutes. During an Oracle RAC failover, there will

be no Oracle Directory Integration Platform down time if one Oracle RAC instance is running at all times.

```
RuntimeException:
[2008-11-21T00:11:10.915-08:00] [wls_ods] [ERROR] []
[org.quartz.impl.jdbcjobstore.JobStoreTX] [tid: 25] [userId: <anonymous>]
[ecid: 0000Hqy69UiFW7V6u3FCEH199aj0000009,0] [APP: DIP] ClusterManager: Error
managing cluster: Failed to obtain DB connection from data source
'schedulerDS': java.sql.SQLException: Could not retrieve datasource via JNDI
url 'jdbc/schedulerDS' java.sql.SQLException: Cannot obtain connection:
driverURL = jdbc:weblogic:pool:schedulerDS, props =
{EmulateTwoPhaseCommit=false, connectionPoolID=schedulerDS,
jdbcTxDataSource=true, LoggingLastResource=false,
dataSourceName=schedulerDS}.[[
Nested Exception: java.lang.RuntimeException: Failed to setAutoCommit to true
for pool connection

AuthenticationException while connecting to OID:
[2008-11-21T00:12:08.812-08:00] [wls_ods] [ERROR] [DIP-10581] [oracle.dip]
[tid: 11] [userId: <anonymous>] [ecid: 0000Hqy6m54FW7V6u3FCEH199apO000000,0]
[APP: DIP] DIP was not able to get the context with the given details {0}[[
javax.naming.AuthenticationException: [LDAP: error code 49 - Invalid
Credentials]
```

Most exceptions are related to the scheduler or LDAP, for example:

- Could not retrieve datasource via JNDI url 'jdbc/schedulerDS' java.sql.SQLException
- javax.naming.AuthenticationException: [LDAP: error code 49 - Invalid Credentials]

Node Manager Fails to Start

If the Node Manager fails to start, ensure that you have copied the `nodemanager.domains` file from `ODIPHOST1` to `ODIPHOST2`:

```
WL_HOME/common/nodemanager/nodemanager.domains
```

Error Messages May Appear After Starting Node Manager

If you see the following error message after starting Node Manager, follow the procedure described after the error message:

```
<Dec 15, 2008 8:40:05 PM> <Warning> <Uncaught exception in server handler:
javax.net.ssl.SSLKeyException: [Security:090482]BAD_CERTIFICATE alert was
received from stbee21.example.com - 152.68.64.2155. Check the peer to
determine why it rejected the certificate chain (trusted CA configuration,
hostname verification). SSL debug tracing may be required to determine the
exact reason the certificate was rejected.> javax.net.ssl.SSLKeyException:
[Security:090482]BAD_CERTIFICATE alert was received from stbee21.example.com -
152.68.64.215. Check the peer to determine why it rejected the certificate chain
(trusted CA configuration, hostname verification). SSL debug tracing may be
required to determine the exact reason the certificate was rejected.
```

1. If you have not already done so, click **Lock & Edit** in the Change Center of the Administration Console.
2. In the left pane of the Console, expand **Servers** and **AdminServer (admin)**.
3. Select the **Configuration > SSL > Advanced Link**.
4. Select **None** for **Hostname Verification**.
5. Click **Save** to save the setting.

6. To activate these changes, in the Change Center of the Administration Console, click **Activate Changes**.
7. Restart all servers.

(Optional) Enter an example to illustrate your reference here.

1. If you have not already done so, click **Lock & Edit** in the Change Center of the Administration Console.
2. In the left pane of the Console, expand **Servers** and the name of the server that is running in ADMIN mode.
3. Select the **Control > Start/Stop** tab.
4. Select the name of the server.
5. Click **Resume**.
6. Select **Yes** to resume servers.

Configuration Changes Do Not Automatically Propagate to All Oracle Directory Integration Platform Instances in a Highly Available Topology

When you change the configuration of one Oracle Directory Integration Platform instance in a high availability topology, the configuration change does not propagate automatically to all Oracle Directory Integration Platform instances in the topology.

Use the `manageDIPServerConfig` tool to make configuration change to all Oracle Directory Integration Platform instances in the topology, ensuring the same configuration across all Oracle Directory Integration Platform instances.

See manageDIPServerConfig Utility in *Oracle Fusion Middleware Administering Oracle Directory Integration Platform*.

An Operation Cannot Be Completed for Unknown Errors Message Appears

The following error message may appear intermittently when you use the `manageSyncProfiles` command:

```
OPERATION CANNOT BE COMPLETED FOR UNKNOWN ERRORS
```

If you see this error message, start and stop the Managed Server (`wls_ods1` or `wls_ods2`). If the problem persists, repeat the copy method on the second node.

About Starting and Stopping Oracle Directory Services Components

To start and stop Oracle Directory Services Components components, see Starting and Stopping Components in *Oracle Fusion Middleware Administering Oracle Fusion Middleware*.

8

Uninstalling or Reinstalling Oracle Internet Directory

Follow the instructions in this section to uninstall or reinstall Oracle Internet Directory.

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.

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 the software. 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 a directory outside of the `ORACLE_HOME`.

   Do not attempt to run the deinstall executable from the `ORACLE_HOME` location as this can cause the deinstallation to fail due to files have already been deleted and no longer accessible.

2. Enter the following command:

   (UNIX) `$ORACLE_HOME/oui/bin/deinstall.sh`

   (Windows) `%ORACLE_HOME%\oui\bindeinstall.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 the product you want to remove 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 Internet Directory 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.

Table 8-2 describes the screens in the Uninstall Wizard. For information, click **Help** on the screen.

Table 8-2 Uninstall Wizard Screens and Descriptions

Screen	Description
Welcome	Introduces you to the product Uninstall Wizard.
Dinstallation 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 Linux operating systems, enter the following commands:

```
cd /home/Oracle/product
rm -rf ORACLE_HOME
```

On Windows operating systems, 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 Linux operating systems, if your Domain home directory is `/home/Oracle/config/domains/oid_domain`, enter the following command:

   ```
   cd /home/Oracle/config/domains

   rm -rf oid_domain
   ```

   On Windows operating systems, if your Domain home directory is `C:\Oracle\Config\domains\oid_domain`, use a file manager window and navigate to the `C:\Oracle\Config\domains` directory. Right-click on the `oid_domain` folder and select **Delete**.
2. Manually remove your Application home directory. For example:

   On Linux operating systems, if your Application home directory is `/home/Oracle/config/applications/oid_domain`, enter the following commands:

   ```
   cd /home/Oracle/config/applications

   rm -rf oid_domain
   ```

   On Windows operating systems, if your Application home directory is `C:\Oracle\Config\applications\oid_domain`, use a file manager window and navigate to the `C:\Oracle\Config\applications` directory. Right-click on the `oid_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 `oid_domain`, find the following line and remove it:

   ```
   <domain location="/home/Oracle/config/domains/oid_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 an unsupported JDK version 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. The supported JDK version for this release is jdk17.0.12 and it 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.

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/jdk17.0.12`.

- 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.

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:

(Linux) *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 jdk17.0.12.

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:

   (Linux) *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:

   (Linux) *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:

   (Linux) *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 jdk17.0.12.

Updating the JDK Location in an Existing Domain Home

You must search the references to the current JDK manually, and replace those instances with the location of the new JDK.

You can use the `grep` or `findstr` commands to search for the jdk-related references.

You'll likely be required to update the location of JDK in the following three files:

(Linux) *DOMAIN_HOME*`/bin/setNMJavaHome.sh`

(Windows) *DOMAIN_HOME*`\bin\setNMJavaHome.cmd`

(Linux) *DOMAIN_HOME*`/nodemanager/nodemanager.properties`

(Windows)*DOMAIN_HOME*`\nodemanager\nodemanager.properties`

(Linux) *DOMAIN_HOME*`/bin/setDomainEnv.sh`

(Windows)*DOMAIN_HOME*`\bin\setDomainEnv.cmd`