Oracle® Fusion Middleware

Upgrading Oracle Identity Manager

14c (14.1.2.1.0)
F85547-01
March 2025

Oracle Fusion Middleware Upgrading Oracle Identity Manager, 14c (14.1.2.1.0)

F85547-01

Primary Author: Oracle Corporation

Contents

4 Upgrading Oracle Identity Manager Highly Available Environments

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

Preface

This document describes how to upgrade Oracle Identity Manager to 14c (14.1.2.1.0).

- Audience
  This document is intended for system administrators who are responsible for installing, maintaining, and upgrading Oracle Identity Manager.
- Documentation Accessibility
- Diversity and Inclusion
- Related Documents
- Conventions
  Learn about the conventions used in this document.

Audience

This document is intended for system administrators who are responsible for installing, maintaining, and upgrading Oracle Identity Manager.

It is assumed that readers have knowledge of the following:

- Oracle Fusion Middleware system administration and configuration.
- Configuration parameters and expected behavior of the system being upgraded.

Documentation Accessibility

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

Access to Oracle Support

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

Diversity and Inclusion

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

Related Documents

Refer to the Oracle Fusion Middleware Library for additional information.

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

Conventions

Learn about the conventions used in this document.

This document uses the following text conventions:

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

1

Introduction to Upgrading Oracle Identity Manager to 14c (14.1.2.1.0)

Before you begin, review all introductory information to understand the standard upgrade topologies and upgrade paths for Oracle Identity Manager 14c (14.1.2.1.0).

Note:

- The product Oracle Identity Manager is referred to as Oracle Identity Manager (OIM) and Oracle Identity Governance (OIG) interchangeably in the guide.
- Oracle recommends that you perform the upgrade as documented in this guide. If you require design/architectural changes (for example: changing the directory structure), complete them as separate steps during the post-upgrade validations.
- For general information about Fusion Middleware upgrade planning and other upgrade concepts and resources, see the following sections in *Planning an Upgrade of Oracle Fusion Middleware*:
  - Planning an Upgrade to Oracle Fusion Middleware 14c (14.1.2.1.0)
  - Understanding In-Place versus Out-of-Place Upgrades
  - Understanding the Basic Upgrade Tasks

The following topics describe the concepts related to upgrading Oracle Identity Manager:

- About the Starting Points for a Oracle Identity Manager Upgrade
  The only supported starting point for an upgrade to Oracle Identity Manager 14c (14.1.2.1.0) is the Oracle Identity Manager 12c (12.2.1.4.0) release.
- About the Oracle Identity Manager Upgrade Scenarios
  The steps to upgrade Oracle Identity Manager to 14c (14.1.2.1.0) depend on the existing production topology.
- About the New Features for Oracle Identity Manager 14c (14.1.2.1.0)
  Several changes have been made to Oracle Identity Manager between 12c (12.2.1.4.0) and 14c (14.1.2.1.0).
- About Upgrade Restrictions
  If you are using two or more Oracle Fusion Middleware products of the same or different versions in a single, supported, Oracle Fusion Middleware configuration, you must consider the interoperability and compatibility factors before planning the upgrade.
- Terminology Used in this Guide
  For consistency, the following terminology is used in this guide.
- How to Use This Guide
  This guide covers various upgrade scenarios.

About the Starting Points for a Oracle Identity Manager Upgrade

The only supported starting point for an upgrade to Oracle Identity Manager 14c (14.1.2.1.0) is the Oracle Identity Manager 12c (12.2.1.4.0) release.

Upgrading From 12c (12.2.1.4)

You can upgrade a 12c (12.2.1.4.0) release to 14c (14.1.2.1.0) by using one of the following methods:

- In-place upgrade: See In-Place Upgrade of Oracle Identity Manager.
- Out-of-place cloned upgrade: See Out-of-Place Cloned Upgrade of Oracle Identity Manager.

.

The upgrade procedures in this guide explain how to upgrade an existing Oracle Identity Manager to Oracle Identity Manager 14c (14.1.2.1.0). If your domain contains other components, you will have to upgrade those components as well.

About the Oracle Identity Manager Upgrade Scenarios

The steps to upgrade Oracle Identity Manager to 14c (14.1.2.1.0) depend on the existing production topology.

Oracle Identity Manager can be deployed in a number of different ways. This upgrade documentation provides instructions for the common deployment topologies. However, it can be used as a guide for the less common deployment topologies as well.

Your actual topology may vary, but the topologies described here provide an example that can be used as a guide to upgrade other similar Oracle Identity Manager topologies.

> **Note:**
>
> For additional information about the upgrade process and planning resources to ensure your upgrade is successful, see Preparing to Upgrade in *Planning an Upgrade of Oracle Fusion Middleware*.

You can upgrade the following topologies or deployments using the procedure described in this guide:

- Single node environments
- Highly available (multinode) environments

About the New Features for Oracle Identity Manager 14c (14.1.2.1.0)

Several changes have been made to Oracle Identity Manager between 12c (12.2.1.4.0) and 14c (14.1.2.1.0).

To understand what's new in general in 14c (14.1.2.1.0), see New and Changed Features in *Understanding Oracle Fusion Middleware*.

If your environment includes Oracle WebLogic Server with Oracle ADF, see Key Differences Between Application Developer 11g and Infrastructure 14c (14.1.2.1.0).

For more information about Oracle Identity Governance 14c (14.1.2.1.0), refer to the following topics in the *Administering Oracle Identity Governance*:

- New and Changed Features for 14c (14.1.2.1.0)
- What is Oracle Identity Governance?
- What are the Different Modes of Oracle Identity Governance?

About Upgrade Restrictions

If you are using two or more Oracle Fusion Middleware products of the same or different versions in a single, supported, Oracle Fusion Middleware configuration, you must consider the interoperability and compatibility factors before planning the upgrade.

Interoperability

In the context of Oracle Fusion Middleware products, Interoperability is defined as the ability of two Oracle Fusion Middleware products or components of the same version (or release) to work together (interoperate) in a supported Oracle Fusion Middleware configuration. Specifically, interoperability applies when the first 4 digits of the release or version number are the same. For example, Oracle Fusion Middleware 14c (14.1.2.1.0) components are generally interoperable with other 14c (14.1.2.1.0) components. See Interoperability with Oracle Identity Management Products.

> **Note:**
>
> Exporting and importing OAM policies from other releases by using tools such as exportPolicy, importPolicy, and so on, is not certified. An upgrade is the only supported path to move policies from one release to another.

Compatibility

In the context of Oracle Fusion Middleware products, Compatibility is defined as the ability of two Oracle Fusion Middleware components of different versions (or releases) to interoperate.

For a list of products and features available in Oracle Fusion Middleware Release 14.1.2.1.0, see Products and Features Available in Oracle Fusion Middleware 14c (14.1.2.1.0) in *Understanding Interoperability and Compatibility*.

Terminology Used in this Guide

For consistency, the following terminology is used in this guide.

Table 1-1 Terminology

Information	Example Value	Description
`JAVA_HOME`	`/home/Oracle/Java/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.

Table 1-1 (Cont.) Terminology

Information	Example Value	Description
Database port	`1521`	Port number that the database listens on. The default Oracle database listen port is `1521`.
Database service name	`orcl.exampledomain`	Oracle databases require a unique service name. The default service name is `orcl`.
DBA username	`FMW`	Name of user with database administration privileges. The default DBA user on Oracle databases is `SYS`.
DBA password	*<dba_password>*	Password of the user with database administration privileges.
ORACLE_HOME	`/u01/app/fmw/ORACLE_HOME`	12c directory in which you will install your software. This directory will include Oracle Fusion Middleware Infrastructure and Oracle Identity Manager, as needed.
Console port	`7001`	Port for Oracle WebLogic Server and Oracle Identity Manager consoles.
DOMAIN_HOME	`/home/Oracle/config/domains/idm_domain`	Location in which your domain data is stored. **Note:** This is the domain where the primary Administration server is configured.
APPLICATION_HOME	`/home/Oracle/config/applications/idm_domain`	Location in which your application data is stored.
Administrator user name for your WebLogic domain	`weblogic`	Name of the user with Oracle WebLogic Server administration privileges. The default administrator user is `weblogic`.
Administrator user password	*<admin_password>*	Password of the user with Oracle WebLogic Server administration privileges.
RCU	*ORACLE_HOME*`/oracle_common/bin`	Path to the Repository Creation Utility (RCU).
RCU schema prefix	`oim`	Prefix for names of database schemas used by Oracle Identity Manager.
RCU schema password	*<rcu_password>*	Password for the database schemas used by Oracle Identity Manager.
Configuration utility	*ORACLE_HOME*`/oracle_common/common/bin`	Path to the Configuration Wizard for domain creation and configuration.

How to Use This Guide

This guide covers various upgrade scenarios.

Depending on your existing deployment, refer to the respective topics for upgrading Oracle Identity Manager to 14c (14.1.2.1.0):

- **In-Place Upgrade**
  - **Single Node Environments**: For upgrading a single node Oracle Identity Manager (OIM) setup, see Upgrading Oracle Identity Manager Single Node Environments.
  - **Multi-node or Highly Available Environments**: For upgrading a multi-node Oracle Identity Manager setup, see Upgrading Oracle Identity Manager Highly Available Environments.
- **Out-of-Place Upgrade**: For instructions to upgrade out-of-place, see Performing an Out-of-Place Upgrade of Oracle Identity Manager
- **Out-of-Place Cloned Upgrade**: For instructions to perform an out-of-place cloned upgrade, see Performing an Out-of-Place Cloned Upgrade of Oracle Identity Manager.

> **Note:**
>
> Before you begin the upgrade, ensure that you review the Pre-Upgrade Requirements and perform the necessary pre-upgrade tasks.

2

Pre-Upgrade Requirements

Before you begin to upgrade Oracle Identity Manager 14c (14.1.2.1.0), you must perform pre-upgrade tasks such as backing up, creating a replica of your current environment, and verifying that your system meets certified requirements.

- Oracle Fusion Middleware Pre-Upgrade Checklist
  Perform the tasks in this checklist before you begin any upgrade to ensure you have a successful upgrade and limited downtime.
- Creating a Complete Backup
  Before you start an upgrade, back up all system-critical files, including the Oracle home, Domain home, and databases that host your Oracle Fusion Middleware schemas.
- Generating and Analyzing Pre-Upgrade Report for Oracle Identity Manager
  Run the pre-upgrade report utility before you begin the upgrade process for Oracle Identity Manager, and address all of the issues using the solution provided in the report.
- Verifying Certification and System Requirements
  Review the certification matrix and system requirements documents to verify that your environment meets the necessary requirements for installation. You may be required to upgrade your operating system, hardware or other software packages.
- Verify the Database User for the WLSSchemaDataSource Data Source
  This step is required if your existing domain has a `WLSSchemaDataSource` data source.
- Creating a Non-SYSDBA User to Run the Upgrade Assistant
  To run the Upgrade Assistant, Oracle recommends that you create a non-SYSDBA user called `FMW`, within your PDB. This user has the privileges required to modify schemas, but does not have full administrator privileges.
- Identifying Existing Schemas Available for Upgrade
  This optional step can be used before an upgrade to query the schema version registry table. This table contains schema information such as the schema owner, version number, component name and ID, date of creation and modification, and custom prefixes.
- Shutting Down the Node Managers
  Ensure that you have shut down all the local and remote Node Managers before starting the upgrade process.

Oracle Fusion Middleware Pre-Upgrade Checklist

Perform the tasks in this checklist before you begin any upgrade to ensure you have a successful upgrade and limited downtime.

Upgrades are performed while the servers are down. This checklist identifies important and often time-consuming pre-upgrade tasks that you can perform before the upgrade to limit your downtime. The more preparation you do before you begin the upgrade process, the less time you will spend offline.

> **Note:**
>
> The pre-upgrade procedures you perform will depend on the configuration of your existing system, the components you are upgrading, and the environment you want to create at the end of the upgrade and configuration process. Complete only those tasks that apply to your configurations or use cases.

Table 2-1 Tasks to Perform Before You Upgrade to Oracle Fusion Middleware 14c (14.1.2.1.0)

Task	Description
Required Create a complete backup of your existing environment.	Back up all system-critical files and database(s) that contain any schemas that are to be upgraded. If the upgrade fails, you must restore your pre-upgrade environment and begin the upgrade again. See Creating a Complete Backup. • Ensure that your backup includes the schema version registry table. See Backing Up the Schema Version Registry Table. • If you modified any of the startup scripts in your existing domain, you will need to copy them to temporary directory location (outside of the existing domain) during the upgrade and redeploy them after the upgrade. See Maintaining Customized Domain and Environment Settings.

Table 2-1 (Cont.) Tasks to Perform Before You Upgrade to Oracle Fusion Middleware 14c (14.1.2.1.0)

Task	Description
Required Verify that you are installing and upgrading your product on a supported hardware and software configuration. **Caution:** Do not attempt an upgrade if you are unable to use the latest supported operating system. As with all supported configurations, failure to comply with these requirements may cause your upgrade to fail.	Verify that your hardware and software configurations (including operating systems) are supported by the latest certifications and requirements. Also ensure to use a supported JDK version before you install the 14c (14.1.2.1.0) product distributions. Oracle recommends that you verify this information right before you start the upgrade as the certification requirements are frequently updated. **Note:** • Ensure that you have applied the latest patches to your components before you upgrade. 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*. • Upgrade a component at a time, be it an Oracle component or a dependent component. For example, do not upgrade OUD, OIM, OAM, the operating system, the database, and the hardware all at the same time. See Verifying Certification and System Requirements.
Required Change the database user assigned to the `WLSSchemaDataSource` data source from `<PREFIX>_WLS_RUNTIME` to `<PREFIX>_WLS`.	If the database user for the `WLSSchemaDataSource` data source is assigned to `<PREFIX>_WLS_RUNTIME`, then you must change it to `<PREFIX>_WLS` This step is required only if your existing domain has a `WLSSchemaDataSource` data source. See Verify the Database User for the WLSSchemaDataSource Data Source
Optional Purge any outdated or unused data before you upgrade.	To optimize performance, Oracle strongly recommends that you purge data and objects that will not be used in the upgraded environment. See Purging Unused Data.
Optional Create a Non-SYSDBA user to run the Upgrade Assistant.	Oracle recommends that you create the FMW user to run Upgrade Assistant. User FMW can run the Upgrade Assistant without system administration privileges. See Creating a Non-SYSDBA User to Run the Upgrade Assistant.

Table 2-1 (Cont.) Tasks to Perform Before You Upgrade to Oracle Fusion Middleware 14c (14.1.2.1.0)

Task	Description
Optional Review the list of available schemas.	Query the schema version registry to view schema information. See Identifying Existing Schemas Available for Upgrade.
Required Update the database parameters.	See Updating Database Parameters for Oracle Identity Manager.
Optional Shut down all the local and remote Node Managers before starting the upgrade process.	See Shutting Down the Node Managers.
Required Run the pre-upgrade report utility.	See Generating and Analyzing Pre-Upgrade Report for Oracle Identity Manager

Creating a Complete Backup

Before you start an upgrade, back up all system-critical files, including the Oracle home, Domain home, and databases that host your Oracle Fusion Middleware schemas.

The backup must include the `SYSTEM.SCHEMA_VERSION_REGISTRY$` table so that you can restore the contents back to its pre-upgrade state if the upgrade fails.

> **Note:**
>
> The Upgrade Assistant Prerequisites screen prompts you to acknowledge that backups have been performed before you proceed with the actual upgrade. However, the Upgrade Assistant does not verify that a backup has been created.

See:

- Backing Up Your Environment in *Administering Oracle Fusion Middleware*
- Upgrading and Preparing Your Oracle Databases for 14c (14.1.2.1.0) in *Planning an Upgrade of Oracle Fusion Middleware*
- Oracle Database Documentation for information about upgrading the Oracle Database 19*c* and 23ai.
- Backing Up the Schema Version Registry Table
  Your system backup must include the `SYSTEM.SCHEMA_VERSION_REGISTRY` table or the `FMWREGISTRY.SCHEMA_VERSION_REGISTRY` table.
- Maintaining Customized Domain and Environment Settings
  If you have modified any domain-generated, server startup scripts, or configuration files in your pre-upgrade environment, it is important to note that these changes are overwritten during the installation, and reconfiguration operations.

Backing Up the Schema Version Registry Table

Your system backup must include the `SYSTEM.SCHEMA_VERSION_REGISTRY` table or the `FMWREGISTRY.SCHEMA_VERSION_REGISTRY` table.

> **Note:**
>
> This step is only required for managed or collocated domains. Standalone domains will not have this table.

Each Fusion Middleware schema has a row in the `SYSTEM.SCHEMA_VERSION_REGISTRY` table. If you run the Upgrade Assistant to update an existing schema and it does not succeed, you must restore the original schema before you can try again. Before you run the Upgrade Assistant, make sure you back up your existing database schemas and the schema version registry.

> **Note:**
>
> Before you upgrade a schema using the Upgrade Assistant, you must perform a complete database backup. During the upgrade, you are required to acknowledge that backups have been performed.

Maintaining Customized Domain and Environment Settings

If you have modified any domain-generated, server startup scripts, or configuration files in your pre-upgrade environment, it is important to note that these changes are overwritten during the installation, and reconfiguration operations.

Oracle recommends you to take a backup of the the customized files to a shared library location. In case of any failure or issues during the upgrade process, you can restore these files, if required.

Every domain installation includes dynamically-generated domain and server startup scripts, such as `setDomainEnv`. These files are replaced by newer versions during the installation and upgrade process.

For example, if you want to customize server startup parameters that apply to all servers in a domain, you can create a file called `setUserOverridesLate.cmd` (Windows) or `setUserOverridesLate.sh` (UNIX) and configure it to add custom libraries to the WebLogic Server classpath, specify additional command-line options for running the servers, or specify additional environment variables. When using the `pack` and `unpack` commands, any custom settings that you add to this file are preserved during the domain upgrade operation and are carried over to the remote servers.

For an example of startup customizations in the `setUserOverridesLate` script, see Customizing Server Parameters with the setUserOverridesLate Script in *Enterprise Deployment Guide for Oracle WebCenter Portal*.

Generating and Analyzing Pre-Upgrade Report for Oracle Identity Manager

Run the pre-upgrade report utility before you begin the upgrade process for Oracle Identity Manager, and address all of the issues using the solution provided in the report.

The pre-upgrade report utility analyzes your existing Oracle Identity Manager environment, and provides information about the mandatory prerequisites that you must complete before you begin the upgrade.

> **Note:**
>
> It is important to address all of the issues listed in the pre-upgrade report before you proceed with the upgrade, as the upgrade might fail if the issues are not resolved.
>
> Ensure that the Database and the 12.2.1.4.0 Oracle Identity Manager servers are up and running before you run the pre-upgrade report utility.

- Obtaining the Pre-Upgrade Report Utility
  Download the pre-upgrade report utility for Oracle Identity Manager from Oracle Technology Network (OTN).
- Generating the Pre-Upgrade Report
  Generate the pre-upgrade report before you start the upgrade process for Oracle Identity Manager, and resolve any issues listed in the report.
- Analyzing the Pre-Upgrade Report
  After you generate the pre-upgrade report for Oracle Identity Manager, review each of the reports, and perform all of the tasks described in them. If you do not perform the mandatory tasks described in the report, the upgrade might fail.

Obtaining the Pre-Upgrade Report Utility

Download the pre-upgrade report utility for Oracle Identity Manager from Oracle Technology Network (OTN).

The utility is available in a zip file named `PreUpgradeReport.zip` at the following location Identity & Access Management Downloads or from My Oracle Support:
My Oracle Support document ID 3063747.1

Generating the Pre-Upgrade Report

Generate the pre-upgrade report before you start the upgrade process for Oracle Identity Manager, and resolve any issues listed in the report.

To generate the pre-upgrade report for Oracle Identity Manager, complete the following steps on your Administration server host machine:

1. Create a directory at any location and extract the contents of `PreUpgradeReport.zip` in the new directory.
2. Create a directory in which to generate the pre-upgrade reports. For example, create a directory named `OIM_preupgrade_reports`.
3. Go to the directory where you extracted `PreUpgradeReport.zip` and open the `preupgrade_report_input.properties` file in a text editor. Update the properties file with the appropriate values for the parameters listed in Table 2-2

Table 2-2 Parameters to be Specified in the preupgrade_report_input.properties File

Parameter	Description
`oim.targetVersion`	Specify the target version of the Oracle Identity Manager, that is, 14c (14.1.2.1.0).
`oim.jdbcurl`	Specify the JDBC URL for Oracle Identity Manager in one of the following formats: *`host:port/service_name`* or *`host:port:sid`*
`oim.oimschemaowner`	Specify the name of the OIM schema owner. For example, *DEV_OIM*.
`oim.mdsjdbcurl`	Specify the MDS JDBC URL in the one of the following formats: *`host:port/service_name`* or *`host:port:sid`*
`oim.mdsschemaowner`	Specify the name of the MDS schema owner. For example, *DEV_MDS*.
`oim.databaseadminname`	Specify the user with DBA privilege. For example, `sys` as `sysdba`.
`oim.outputreportfolder`	Specify the absolute path to the directory where you want the reports to be generated (`OIM_preupgrade_reports`).Ensure that this directory has read and write permissions.
`oim.mwhome`	Specify the absolute path to the Middleware home. For example: `/Oracle/Middleware`
`oim.oimhome`	Specify the absolute path to the existing OIM home. For example: `/Oracle/Middleware/idm`
`oim.javahome`	Specify the absolute path to the Java home. Ensure that you point to JAVA 8.
`oim.wlshome`	Specify the absolute path to the WebLogic Server home. For example: `/Oracle/Middleware/wlserver`
`oim.domain`	Specify the absolute path to the Oracle Identity Manager domain home. For example: `/Oracle/Middleware/user_projects/domains/IAMGovernanceDomain`

4. Run the following command from the location where you extracted the contents of `PreUpgradeReport.zip`:
   - On UNIX:
     `sh generatePreUpgradeReport.sh`
   - On Windows:
     `generatePreUpgradeReport.bat`
5. Provide the details when the following are prompted:

- **OIM Schema Password**: Enter the password of the Oracle Identity Manager (OIM) schema.
- **MDS Schema Password**: Enter the password of the Metadata Services (MDS) schema.
- **DBA Password**: Enter the password of the Database Administrator.

6. The reports are generated as HTML pages at the location you specified for the parameter `oim.outputreportfolder` in the `preupgrade_report_input.properties` file. The logs are stored in the log file `preUpgradeReport<time>.log` in the folder logs at the same location.

Analyzing the Pre-Upgrade Report

After you generate the pre-upgrade report for Oracle Identity Manager, review each of the reports, and perform all of the tasks described in them. If you do not perform the mandatory tasks described in the report, the upgrade might fail.

Table 2-3 Pre-Upgrade Reports Generated for Oracle Identity Manager

Report Name	Description and Action Item
`MDS Back-up of source environment`	This report lists the details regarding the MDS backup taken prior to upgrade.
`Customized Notification Templates status on source environment`	This report lists customized out-of-the-box (OOTB) notification templates. These customizations will be overwritten with OOTB values during upgrade. **Note:** This report is generated only if there are any discrepancies found.
`Status of Domain Configuration`	This report lists the applications (if any) that are in stage mode.
`Authorization Policy Back-up of source environment`	This report lists the details regarding the Oracle Identity Manager authorization policy backup taken prior to upgrade.
`Copy Custom UI WAR from source environment`	This report reminds you to copy the custom UI war from the previous Middleware home to the new Middleware home, to get the UI customizations after upgrade.

Table 2-3 (Cont.) Pre-Upgrade Reports Generated for Oracle Identity Manager

Report Name	Description and Action Item
`Status of Database Vault Configuration`	This is a conditional report. If database vault is enabled on source setup, then this report is created. This report displays information related to database vault settings. **Note:** This report is generated only if there are any discrepancies found.

Verifying Certification and System Requirements

Review the certification matrix and system requirements documents to verify that your environment meets the necessary requirements for installation. You may be required to upgrade your operating system, hardware or other software packages.

Note:

When checking the certification, system requirements, and interoperability information, be sure to check specifically for any operating system requirements. It is important for you to download software specifically designed for your operating system environment, explicitly.

WARNING:

Make sure that your current environment has been patched to the latest patch set *before* you begin the upgrade. Certifications are based on fully patched environments, unless stated otherwise.

- Verify Your Environment Meets Certification Requirements
  Oracle has tested and verified the performance of your product on all certified systems and environments. Make sure that you are installing your product on a supported hardware and software configuration.
- Verify System Requirements and Specifications
  It is important to use both the System Requirements and Specifications document and the Oracle Fusion Middleware Certification Matrix to verify that the system requirements such as disk space, available memory, specific platform packages and patches, and other operating system-specific items are met.
- Verify That the Database Hosting Oracle Fusion Middleware is Supported
  You must have a supported Oracle database configured with the required schemas before you run Oracle Fusion Middleware 14c (14.1.2.1.0).

- Verify That the JDK Is Certified for This Release of Oracle Fusion Middleware
  If your JDK is not supported, or you do not have a JDK installed, you must download the required Java SE JDK before you begin.

Verify Your Environment Meets Certification Requirements

Oracle has tested and verified the performance of your product on all certified systems and environments. Make sure that you are installing your product on a supported hardware and software configuration.

Whenever new certifications occur, they are added to the appropriate certification document right away. New certifications can occur at any time, and for this reason the certification documents are kept outside of the documentation libraries and are available on Oracle Technical Resources. See the Certification Matrix for 14c (14.1.2.1.0).

Verify System Requirements and Specifications

It is important to use both the System Requirements and Specifications document and the Oracle Fusion Middleware Certification Matrix to verify that the system requirements such as disk space, available memory, specific platform packages and patches, and other operating system-specific items are met.

Use the Oracle Fusion Middleware System Requirements and Specifications document to verify that the requirements of the Oracle Fusion Middleware Certification matrix are met. For example, if the Certification Matrix indicates that your product is certified for installation on 64-Bit Oracle Linux 8, the System Requirements and Specifications document should be used to verify that your Oracle Linux 8 system has met the required minimum specifications such as disk space, available memory, specific platform packages and patches, and other operating system-specific items. This document is updated as needed and resides outside of the documentation libraries on the Oracle Technology Network (OTN).

> **Note:**
>
> Do not attempt an upgrade if you are unable to meet the minimum system requirements.

Specifically, you can use the Oracle Fusion Middleware System Requirements and Specifications document to verify the following:

- Processor Requirements
- Java Development Kit (JDK) Requirements
- General Memory and Disk Space Requirements
- Product-Specific Memory and Disk Space Requirements
- Network Requirements
- UNIX Operating System Requirements
- Windows Operating Systems Requirements
- Virtualization Requirements
- Database Requirements

What if my operating system is not supported?

If you are running your environment on an unsupported operating system, you will need to create a supported environment before you begin your upgrade. Do not attempt an upgrade on an unsupported operating system.

Use the migration steps for your environment.

Verify That the Database Hosting Oracle Fusion Middleware is Supported

You must have a supported Oracle database configured with the required schemas before you run Oracle Fusion Middleware 14c (14.1.2.1.0).

Review the Fusion Middleware database requirements before starting the upgrade to ensure that the database hosting Oracle Fusion Middleware is supported and has sufficient space to perform an upgrade. See the Certification Matrix for 14c (14.1.2.1.0).

> **Note:**
>
> If your database version is no longer supported, you must upgrade to a supported version before starting an upgrade.

- Updating Database Parameters for Oracle Identity Manager
  You need to verify and update a few database parameters before upgrading the Oracle Identity Manager to 14c (14.1.2.1.0).

Updating Database Parameters for Oracle Identity Manager

You need to verify and update a few database parameters before upgrading the Oracle Identity Manager to 14c (14.1.2.1.0).

Complete the following steps:

1. Connect to the database by using an account that has Oracle DBA privileges, and run the commands in this procedure from SQL*Plus.
2. To verify the value for the database parameter `max_string_size`, run the following command:

   ```
   SQL> SELECT value FROM v$parameter WHERE name='max_string_size';
   ```

3. If the value returned is:
   - **STANDARD**: Skip the rest of the steps in this procedure and go to the next procedure to continue with the upgrade.
   - **EXTENDED**: Continue with **step 4**.
4. Login as an OIM database user and then run the following command to find columns with size more than 4000 characters:

   ```
   SQL> SELECT table_name, column_name, data_length FROM user_tab_columns
   WHERE data_length>4000;
   ```

5. If any rows are listed, either trim the corresponding column data to 4000 characters or remove the rows.

> **Note:**
>
> If required, take backup of the listed rows in a new table.

6. Reset all the columns sizes found in step 4 to 4000 characters.
   As an OIM database user, run the following command:

   ```
   SQL> ALTER TABLE <table_name> MODIFY <column_name> VARCHAR2(4000);
   ```

7. On the columns whose length was modified to more than 4000 characters, drop any existing index.
8. As an OIM database user, run the following command to verify that there no more columns with size more than 4000:

   ```
   SQL> SELECT table_name, column_name, data_length FROM user_tab_columns
   WHERE data_length>4000;
   ```

9. If required, gather table and index stats for the identified columns.

For more information, see Monitoring Oracle Identity Governance Performance.

Verify That the JDK Is Certified for This Release of Oracle Fusion Middleware

If your JDK is not supported, or you do not have a JDK installed, you must download the required Java SE JDK before you begin.

Refer to the Oracle Fusion Middleware Supported System Configurations information on the Oracle Technology Network (OTN) to verify that the JDK you are using is supported.

If your JDK is not supported, or you do not have a JDK installed, you must download the required Java SE JDK, from the following website:

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

Make sure that the JDK is installed outside of the Oracle home. The Oracle Universal Installer validates that the designated Oracle home directory is empty, and the install does not progress until an empty directory is specified. If you install JDK under Oracle home, you may experience issues in future operations. Therefore, Oracle recommends that you use install the JDK in the following directory: `/home/oracle/products/jdk`.

Verify the Database User for the WLSSchemaDataSource Data Source

This step is required if your existing domain has a `WLSSchemaDataSource` data source.

If your domain has the `WLSSchemaDataSource` data source, then you will need to verify which database user is assigned to it. If `<PREFIX>_WLS_RUNTIME` is assigned to it, then you need to change that to `<PREFIX>_WLS` .

This change is necessary due to the following changes:

- The 14c (14.1.2.1.0) Upgrade Assistant uses the information in the `WLSSchemaDataSource` data source,when a domain-based schema upgrade is performed. That upgrade will fail if the `<PREFIX>_WLS` database user is not assigned to the `WLSSchemaDataSource`, or if `<PREFIX>_WLS` is not entered as the "Schema User Name" on the "WLS Schema" page of the Upgrade Assistant.
- Oracle recommends that you use the 12c Oracle WebLogic Administration Console to change the database user to `<PREFIX>_WLS` in the `WLSSchemaDataSource` data source. Doing this will avoid the Upgrade Assistant failure, and also allow the Reconfiguration Wizard to pre-populate fields with the correct values.
- The `<PREFIX>_WLS_RUNTIME` database user is reserved for use with a new `WLSRuntimeSchemaDataSource`, which was introduced in 14c (14.1.2.1.0). This new `WLSRuntimeSchemaDataSource` will be created when the 14c (14.1.2.1.0) Reconfiguration Wizard (reconfig.sh) is used to upgrade the domain.

You can use your Oracle WebLogic 12*c* Administration Console to change the user in the WLSSchemaDataSource from `<PREFIX>_WLS_RUNTIME` to `<PREFIX>_WLS`.

1. Log in the 12*c* (12.2.1.4.0) Administration Console.
2. In the administration console under Domain Structure, expand **Services** (by clicking the **+** next to it). Then click **Data Sources**.
3. If the user in Properties field contains `<PREFIX>_WLS_RUNTIME`, change it to `<PREFIX>_WLS`.
4. Save the change.
5. Use the Change Center to commit the change, if your domain is running in production mode.

Creating a Non-SYSDBA User to Run the Upgrade Assistant

To run the Upgrade Assistant, Oracle recommends that you create a non-SYSDBA user called `FMW`, within your PDB. This user has the privileges required to modify schemas, but does not have full administrator privileges.

> **Note:**
>
> If you run the commands in `cdb`, it fails to create some of the grants successfully.

SYSDBA is an administrative privilege that is required to perform high-level administrative operations such as creating, starting up, shutting down, backing up, or recovering the database. The SYSDBA system privilege is for a fully empowered database administrator. When you connect with the SYSDBA privilege, you connect with a default schema and not with the schema that is generally associated with your user name. For SYSDBA, this schema is SYS. Access to a default schema can be a very powerful privilege. For example, when you connect as user SYS, you have unlimited privileges on data dictionary tables. Therefore, Oracle recommends that you create a non-SYSDBA user to upgrade the schemas. The privileges listed below must be granted to user FMW before starting the Upgrade Assistant.

> **Note:**
>
> The non-SYSDBA user `FMW` is created solely for the purpose of running the Upgrade Assistant. After this step is complete, drop the `FMW` user. The privileges required for running the Upgrade Assistant may change from release to release.
> By default, the `v$xatrans$` table does not exist. You must run the `XAVIEW.SQL` script to create this table before creating the user.
>
> Before creating the user, confirm whether the `v$xatrans$` table was created by a prior upgrade. As a system user, run the following command from `sqlplus`:
>
> ```
> select object_name, owner, object_type from dba_objects where object_name
> like '%XATRANS%'
> ```
>
> If the `v$xatrans$` table was created by a prior upgrade, you will see that the four objects are already available.

In the following example, `password` is the password that you set for the FMW user. When granting privileges, make sure that you specify your actual password.

```
create user FMW identified by password;
grant dba to FMW;
grant execute on DBMS_LOB to FMW with grant option;
grant execute on DBMS_OUTPUT to FMW with grant option;
grant execute on DBMS_STATS to FMW with grant option;
grant execute on sys.dbms_aqadm to FMW with grant option;
grant execute on sys.dbms_aqin to FMW with grant option;
grant execute on sys.dbms_aqjms to FMW with grant option;
grant execute on sys.dbms_aq to FMW with grant option;
grant execute on utl_file to FMW with grant option;
grant execute on dbms_lock to FMW with grant option;
grant select on sys.V_$INSTANCE to FMW with grant option;
grant select on sys.GV_$INSTANCE to FMW with grant option;
grant select on sys.V_$SESSION to FMW with grant option;
grant select on sys.GV_$SESSION to FMW with grant option;
grant select on dba_scheduler_jobs to FMW with grant option;
grant select on dba_scheduler_job_run_details to FMW with grant option;
grant select on dba_scheduler_running_jobs to FMW with grant option;
grant select on dba_aq_agents to FMW with grant option;
grant execute on sys.DBMS_SHARED_POOL to FMW with grant option;
grant select on dba_2pc_pending to FMW with grant option;
grant select on dba_pending_transactions to FMW with grant option;
grant execute on DBMS_FLASHBACK to FMW with grant option;
grant execute on dbms_crypto to FMW with grant option;
grant execute on DBMS_REPUTIL to FMW with grant option;
grant execute on dbms_job to FMW with grant option;
grant select on pending_trans$ to FMW with grant option;
grant select on dba_scheduler_job_classes to FMW with grant option;
grant select on sys.DBA_TABLESPACE_USAGE_METRICS to FMW with grant option;
grant select on SYS.DBA_DATA_FILES to FMW with grant option;
grant select on SYS.V_$ASM_DISKGROUP to FMW with grant option;
grant select on v$xatrans$ to FMW with grant option;
grant execute on sys.dbms_system to FMW with grant option;
grant execute on DBMS_SCHEDULER to FMW with grant option;
```

```
grant select on dba_data_files to FMW with grant option;
grant execute on UTL_RAW to FMW with grant option;
grant execute on DBMS_XMLDOM to FMW with grant option;
grant execute on DBMS_APPLICATION_INFO to FMW with grant option;
grant execute on DBMS_UTILITY to FMW with grant option;
grant execute on DBMS_SESSION to FMW with grant option;
grant execute on DBMS_METADATA to FMW with grant option;
grant execute on DBMS_XMLGEN to FMW with grant option;
grant execute on DBMS_DATAPUMP to FMW with grant option;
grant execute on DBMS_MVIEW to FMW with grant option;
grant select on ALL_ENCRYPTED_COLUMNS to FMW with grant option;
grant select on dba_queue_subscribers to FMW with grant option;
grant execute on SYS.DBMS_ASSERT to FMW with grant option;
grant select on dba_subscr_registrations to FMW with grant option;
grant manage scheduler to FMW;
```

If you are upgrading Oracle Identity Manager (OIM) schema, ensure that the FMW user has the following additional privileges:

```
grant execute on SYS.DBMS_FLASHBACK to fmw with grant option;
grant execute on sys.DBMS_SHARED_POOL to fmw with grant option;
grant execute on SYS.DBMS_XMLGEN to FMW with grant option;
grant execute on SYS.DBMS_DB_VERSION to FMW with grant option;
grant execute on SYS.DBMS_SCHEDULER to FMW with grant option;
grant execute on SYS.DBMS_SQL to FMW with grant option;
grant execute on SYS.DBMS_UTILITY to FMW with grant option;
grant ctxapp to FMW with admin option;
grant execute on SYS.DBMS_FLASHBACK TO FMW with grant option;
grant create MATERIALIZED VIEW to FMW with admin option;
grant all on SCHEMA_VERSION_REGISTRY TO FMW with grant option;
grant create SYNONYM to FMW with admin option;
grant execute on CTXSYS.CTX_ADM to FMW with grant option;
grant execute on CTXSYS.CTX_CLS TO FMW with grant option;
grant execute on CTXSYS.CTX_DDL TO FMW with grant option;
grant execute on CTXSYS.CTX_DOC TO FMW with grant option;
grant execute on CTXSYS.CTX_OUTPUT TO FMW with grant option;
grant execute on CTXSYS.CTX_QUERY TO FMW with grant option;
grant execute on CTXSYS.CTX_REPORT TO FMW with grant option;
grant execute on CTXSYS.CTX_THES TO FMW with grant option;
grant execute on CTXSYS.CTX_ULEXER TO FMW with grant option;
grant create JOB to FMW with admin option;
```

Identifying Existing Schemas Available for Upgrade

This optional step can be used before an upgrade to query the schema version registry table. This table contains schema information such as the schema owner, version number, component name and ID, date of creation and modification, and custom prefixes.

You can let the Upgrade Assistant upgrade all of the schemas in the domain, or you can select individual schemas to upgrade. To help decide, follow these steps to view a list of all the schemas that are available for an upgrade:

1. If you are using an Oracle database, connect to the database by using an account that has Oracle DBA privileges, and run the following from SQL*Plus:

```
SET LINE 120
COLUMN MRC_NAME FORMAT A14
COLUMN COMP_ID FORMAT A20
COLUMN VERSION FORMAT A12
COLUMN STATUS FORMAT A9
COLUMN UPGRADED FORMAT A8
SELECT MRC_NAME, COMP_ID, OWNER, VERSION, STATUS, UPGRADED FROM
SCHEMA_VERSION_REGISTRY WHERE OWNER LIKE UPPER('<PREFIX>_%');
```

2. Examine the report that is generated.

Notes:

- After the upgrade you can generate the report again to see the updated versions of your schemas. If an upgrade was not needed for a schema, the `schema_version_registry` table retains the schema at its pre-upgrade version.
- If your existing schemas are not from a supported version, then you must upgrade them to a supported version before using the 14c (14.1.2.1.0) upgrade procedures. Refer to your pre-upgrade version documentation for more information.
- If you used an OID-based policy store in the earlier versions, make sure to create a new OPSS schema before you perform the upgrade. After the upgrade, the OPSS schema remains an LDAP-based store.
- You can only upgrade schemas for products that are available for upgrade in Oracle Fusion Middleware release 14c (14.1.2.1.0). Do not attempt to upgrade a domain that includes components that are not yet available for upgrade to 14c (14.1.2.1.0).

Shutting Down the Node Managers

Ensure that you have shut down all the local and remote Node Managers before starting the upgrade process.

The Node Managers should remain shut down until you start the WebLogic Administration Server after completing the upgrade. When the WebLogic Administration Server is up and running, start the Node Managers, followed by the Managed Servers.

Part I
In-Place Upgrade of Oracle Identity Manager

You can perform an in-place upgrade of Oracle Identity Manager single node deployments and highly available environments by using the procedures described in this part.

This part contains the following topics:

- Upgrading Oracle Identity Manager Single Node Environments
  You can upgrade Oracle Identity Manager from Release 12c (12.2.1.4.0) to Oracle Identity Governance 14c (14.1.2.1.0) .
- Upgrading Oracle Identity Manager Highly Available Environments
  This chapter describes the process of upgrading an Oracle Identity Manager highly available environment from 12c (12.2.1.4.0) to Oracle Identity Governance 14c (14.1.2.1.0).

3

Upgrading Oracle Identity Manager Single Node Environments

You can upgrade Oracle Identity Manager from Release 12c (12.2.1.4.0) to Oracle Identity Governance 14c (14.1.2.1.0) .

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

Complete the steps in the following topics to perform the upgrade:

- About the Oracle Identity Manager Single Node Upgrade Process
  Review the roadmap for an overview of the upgrade process for Oracle Identity Manager single node deployments.
- Completing the Pre-Upgrade Tasks for Oracle Identity Manager
  Complete the pre-upgrade tasks described in this section before you upgrade Oracle Identity Manager.
- Stopping Servers and Processes
  Before you run the Upgrade Assistant to upgrade the schemas and configurations, you must shut down all the pre-upgrade processes and servers, including the Administration Server, Node Manager (if you have configured Node Manager), and any managed servers.
- Backing up the 12c (12.2.1.4.0) Oracle Home Folder on OIMHOST
  Backup the 12c (12.2.1.4.0) Oracle Home on OIMHOST.
- Installing Product Distributions
  Before beginning your upgrade, download Oracle Fusion Middleware Infrastructure, Oracle SOA Suite, and Oracle Identity Manager distributions on the target system and install them by using the following procedures.
- Running a Pre-Upgrade Readiness Check
  To identify potential issues with the upgrade, Oracle recommends that you run a readiness check before you start the upgrade process. Be aware that the readiness check may not be able to discover all potential issues with your upgrade. An upgrade may still fail, even if the readiness check reports success.
- Upgrading Product Schemas
  After stopping servers and processes, use the Upgrade Assistant to upgrade supported product schemas to the current release of Oracle Fusion Middleware.
- Reconfiguring the Domain
  Run the Reconfiguration Wizard to reconfigure your domain component configurations to 14c (14.1.2.1.0).
- Upgrading Domain Component Configurations
  Use the Upgrade Assistant to upgrade the domain *component* configurations inside the domain to match the updated domain configuration.

- Tuning Application Module for User Interface
  After you successfully upgrade the Oracle Identity Manager middle-tier, tune the Application Module (AM).
- Copying oracle.iam.ui.custom-dev-starter-pack.war from 12c Oracle Home
  You have to manually copy the `oracle.iam.ui.custom-dev-starter-pack.war` file from the backup of 12c (12.2.1.4.0) Oracle Home to 14c (14.1.2.1.0) Oracle home: *ORACLE_HOME*`/idm/server/apps/`.
- Copying Folders to the 14c (14.1.2.1.0) Oracle Home
  When upgrading to 14*c*, you must manually copy some folders to the new Oracle Home, if those folders have file system dependent data.
- Starting Servers and Processes
  After a successful upgrade, start all processes and servers, including the Administration Server and any Managed Servers.
- Verifying the Domain-Specific-Component Configurations Upgrade
  To verify that the domain-specific-component configurations upgrade was successful, sign in to the Remote Console and verify that the version numbers for each upgraded component is 14.1.2.1.0.
- Updating the setDomainEnv.sh File
  For upgrading Oracle Identity Governance (OIG) from 12c (12.2.1.4.0) to 14c (14.1.2.1.0), you need to the delete a property in the `setDomainEnv.sh` file.
- Upgrading Oracle Identity Manager Design Console
  Upgrade the Oracle Identity Manager Design Console after you upgrade the Oracle Identity Manager (OIM) domain component configurations.
- Post-Upgrade Tasks
  After performing the upgrade of Oracle Access Manager to 14c (14.1.2.1.0), you should complete the tasks summarized in this section, if required.

About the Oracle Identity Manager Single Node Upgrade Process

Review the roadmap for an overview of the upgrade process for Oracle Identity Manager single node deployments.

The steps you take to upgrade your existing domain will vary depending on how your domain is configured and which components are being upgraded. Follow only those steps that are applicable to your deployment.

Table 3-1 Tasks for Upgrading Oracle Identity Manager Single Node Environments

Task	Description
Required If you have not done so already, review the introductory topics in this guide and complete the required pre-upgrade tasks.	See: • Introduction to Upgrading Oracle Identity and Access Management to 14c (14.1.2.1.0) • Pre-Upgrade Requirements
Required Shut down the 12c servers. This includes the Administration Server, Managed Servers, Node Manager, and system components such as Oracle HTTP Server. Ensure that the Database is up during the upgrade.	**WARNING**: Failure to shut down your servers during an upgrade may lead to data corruption. See Stopping Servers and Processes.

Table 3-1 (Cont.) Tasks for Upgrading Oracle Identity Manager Single Node Environments

Task	Description
Required Create backup of the existing 12c (12.2.1.4.0) Middleware home folders on OIMHOST	See Backing up the 12c (12.2.1.4.0) Oracle Home Folder on OIMHOST.
Required Run the pre-upgrade report utility and address any detected issues before you begin the upgrade.	The pre-upgrade report utility analyzes your existing Oracle Identity Manager environment, and provides information about the mandatory prerequisites that you must complete before you begin the upgrade. See Generating and Analyzing Pre-Upgrade Report for Oracle Identity Manager
Create a new 14*c* Middleware home location.	The new 14c (14.1.2.1.0) Middleware home location should be on the same host as the production deployment before you begin the upgrade.
Required Install Fusion Middleware Infrastructure 14*c* (14.1.2.0.0), Oracle SOA Suite 14*c* (14.1.2.0.0) and Oracle Identity Manager14c (14.1.2.1.0) in the newly created 14c (14.1.2.1.0) Middleware home.	Install the following products in the prepped 14c (14.1.2.1.0) Middleware home on the same host as the production deployment before you begin the upgrade. • Fusion Middleware Infrastructure 14*c* (14.1.2.0.0) • Oracle SOA Suite14*c* (14.1.2.0.0) • Oracle Identity Manager14c (14.1.2.1.0) It is recommended that you use the simplified installation process to install the products mentioned above, using the quick installer. The quick installer installs the Infrastructure, Oracle SOA Suite, and Oracle Identity and Access Management 14c (14.1.2.1.0) in one go. See Installing Oracle Identity Governance Using Quick Installer in the *Installing and Configuring Oracle Identity and Access Management*. The other option is to install these products separately using their respective installers. See Installing Product Distributions.
Optional Use the Upgrade Assistant to run a pre-upgrade readiness check.	See Running a Pre-Upgrade Readiness Check.
Required Start the Upgrade Assistant to upgrade the 12c database schemas.	See Upgrading Product Schemas.

Table 3-1 (Cont.) Tasks for Upgrading Oracle Identity Manager Single Node Environments

Task	Description
Run the Reconfiguration Wizard to reconfigure your domain component configurations to 14.1.2.1.0.	When you reconfigure a WebLogic Server domain, the following items are automatically updated, depending on the applications in the domain: • WebLogic Server core infrastructure • Domain version See Reconfiguring the Domain **Note:** When the domain reconfiguration process starts, you can't undo the changes that it makes. Before running the Reconfiguration Wizard, **ensure that you have backed up the domain** as covered in the pre-upgrade checklist. If an error or other interruption occurs while running the Reconfiguration Wizard, you must restore the domain by copying the files and directories from the backup location to the original domain directory. This is the only way to ensure that the domain has been returned to its original state before reconfiguration.
Required Run the Upgrade Assistant a second time to upgrade domain component configurations.	See Upgrading Domain Component Configurations. **Note:** The `jce` should use unlimited strength crypto policy.
Optional Copy the `oracle.iam.ui.custom-dev-starter-pack.war` file to the 14c (14.1.2.1.0) Middleware Home. **Note:** This step required only if the file is modified for UI customizations.	See Copying oracle.iam.ui.custom-dev-starter-pack.war to the 14c (14.1.2.1.0) Middleware Home.

Table 3-1 (Cont.) Tasks for Upgrading Oracle Identity Manager Single Node Environments

Task	Description
Optional Copy system-dependent data folders to the 14c (14.1.2.1.0) Oracle Home.	When upgrading to 14c (14.1.2.1.0), you must manually copy some folders to the new Oracle Home, if those folders have file system-dependent data.
Required Start the servers.	See Starting Servers and Processes .
Required Verify that the domain-specific-component configurations is successful.	See Verifying the Domain-Specific-Component Configurations Upgrade.
Optional Perform any required post-upgrade tasks.	See Post-Upgrade Tasks.

Completing the Pre-Upgrade Tasks for Oracle Identity Manager

Complete the pre-upgrade tasks described in this section before you upgrade Oracle Identity Manager.

- Verifying the Memory Settings
  To avoid the memory issues for Oracle Identity Manager, ensure that the memory settings are updated as per the requirements.
- Opening the Non-SSL Ports for SSL Enabled Setup
  If you have an SSL enabled and non-SSL disabled setup, you must open the non-SSL ports for the database before you proceed with the Oracle Identity Manager upgrade.
- Clean Temporary Folder
  Clean the `/tmp` folder on all the Oracle Identity Governance host machines.
- Backing Up the metadata.mar File Manually

Verifying the Memory Settings

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

On Linux, as a `root` user, do the following:

1. Ensure that you set the following parameters in the `/etc/security/limits.conf` or `/etc/security/limits.d` file, to the specified values:

   ```
   FUSION_USER_ACCOUNT soft nofile 32767
   FUSION_USER_ACCOUNT hard nofile 327679
   ```

2. Ensure that you set `UsePAM` to `Yes` in the `/etc/ssh/sshd_config` file.
3. Restart `sshd`.
4. Check the `maxproc` limit and increase it to a minimum of 16384, if needed. Increasing the limit will ensure you do not run into memory issues.

   Use the following command to check the limit:

   ```
   ulimit -u
   ```

If less than 16384, use following command to increase the limit of open files:

```
ulimit -u 16384
```

> **Note:**
>
> You can verify that the limit has been set correctly by reissuing the command `ulimit -u`.

To ensure that the settings persist at reboot, add the following line to the `/etc/security/limits.conf` file or `/etc/security/limits.d` file:

```
oracle hard nproc 16384
```

Where, `oracle` is the install user.

5. Log out (or reboot) and log in to the system again.

Opening the Non-SSL Ports for SSL Enabled Setup

If you have an SSL enabled and non-SSL disabled setup, you must open the non-SSL ports for the database before you proceed with the Oracle Identity Manager upgrade.

Ensure that the database listener is listening on the same TCP port for the database servers that you provided to Upgrade Assistant as parameters. For more information, see Enabling SSL for Oracle Identity Governance DB.

Clean Temporary Folder

Clean the `/tmp` folder on all the Oracle Identity Governance host machines.

As the `/tmp` directory is set against the JVM `java.io.tmpdir` property, any unwanted files in the `/tmp` folder can interfere with OIG upgrade process and might result is MDS corruption.

Backing Up the `metadata.mar` File Manually

After you install the 14c (14.1.2.1.0) binaries in the existing Oracle Home, take a backup of the 14c (14.1.2.1.0)_*ORACLE_HOME*>`/idm/server/apps/oim.ear/metadata.mar` file before the upgrade.

Stopping Servers and Processes

Before you run the Upgrade Assistant to upgrade the schemas and configurations, you must shut down all the pre-upgrade processes and servers, including the Administration Server, Node Manager (if you have configured Node Manager), and any managed servers.

An Oracle Fusion Middleware environment can consist of an Oracle WebLogic Server domain, an Administration Server, multiple managed servers, Java components, system components, and a database used as a repository for metadata. The components may be dependent on each other, so they must be stopped in the correct order.

> **Note:**
>
> The procedures in this section describe how to stop the existing, pre-upgrade 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 stop your pre-upgrade Fusion Middleware environment, navigate to the pre-upgrade domain and follow the steps below:

> **Note:**
>
> It is important that you stop the following servers in the correct order.

Step 1: Stop the Managed Servers

Depending on the method you followed to start the managed servers, follow one of the following methods to stop the WebLogic Managed Server:

Method 1: To stop a WebLogic Server Managed Server not managed by Node Manager:

- (UNIX) *DOMAIN_HOME*/bin/stopManagedWebLogic.sh *managed_server_name admin_url*
- (Windows) *DOMAIN_HOME*\bin\stopManagedWebLogic.cmd *managed_server_name admin_url*

When prompted, enter your user name and password.

Method 2: To stop a WebLogic Server Managed Server by using the Weblogic Console:

- Log into Weblogic console as a `weblogic` Admin.
- Go to **Servers** > **Control** tab.
- Select the required managed server.
- Click **Shutdown**.

Method 3: To stop a WebLogic Server Managed Server using node manager, run the following commands:

```
wls:/offline>nmConnect('nodemanager_username','nodemanager_password',
            'AdminServerHostName','5556','domain_name',
            'DOMAIN_HOME','nodemanager_type')

wls:/offline>nmKill('ManagedServerName')
```

Step 2: Stop the Administration Server

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

Follow one of the these methods to stop the Administration Server:

Method 1: To stop the Administration Server not managed by Node Manager:

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

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

Method 2: To stop the Administration Server by using the Weblogic Console:

- Log into Weblogic console as a `weblogic` Admin.
- Go to **Servers** > **Control** tab.
- Select the required admin server.
- Click **Shutdown**.

Method 3: To stop a WebLogic Server Managed Server using Node Manager, run the following commands:

```
wls:/offline>nmConnect('nodemanager_username','nodemanager_password',
            'AdminServerHostName','5556','domain_name',
            'DOMAIN_HOME','nodemanager_type')

wls:/offline>nmKill('AdminServer')
```

Step 4: Stop Node Manager

To stop Node Manager, run the following command:

```
<DOMAIN_HOME>/bin/stopNodeManager.sh
```

Backing up the 12c (12.2.1.4.0) Oracle Home Folder on OIMHOST

Backup the 12c (12.2.1.4.0) Oracle Home on OIMHOST.

As a backup, copy and rename the 12.2.1.4.0 Oracle home folder on OIMHOST.
For example:

From `/u01/app/fmw/ORACLE_HOME` to `/u01/app/fmw/ORACLE_HOME_old`

> **Note:**
>
> Ensure that you back up any custom configuration. Post upgrade, you will restore these configurations.

Installing Product Distributions

Before beginning your upgrade, download Oracle Fusion Middleware Infrastructure, Oracle SOA Suite, and Oracle Identity Manager distributions on the target system and install them by using the following procedures.

Make sure that you have reviewed the preupgrade checklist and verify that you have installed Java Development Kit (JDK) jdk17.0.12 or later.

> **Note:**
>
> When you install the Oracle Fusion Middleware Release 14c software in preparation for upgrade, you should use the same user account that you used to install and configure the existing, pre-upgrade Oracle Fusion Middleware software. On UNIX operating systems, this ensures that the proper owner and group is applied to new Oracle Fusion Middleware 14c files and directories.

> **Note:**
>
> The 14*c* binaries are installed in a different location from the previous 12*c* binaries. You can install 14*c* binaries before any planned downtime for upgrade.

It is recommended that you use the simplified installation process to install the products mentioned above, using the quickstart installer (`fmw_14.1.2.1.0_idmquickstart.jar`). The quickstart installer installs the Infrastructure, Oracle SOA Suite, and Oracle Identity Manager in one go.

> **Note:**
>
> If you are using Redundant binary locations, ensure that you install the software into each of those redundant locations.

See Installing Oracle Identity Governance Using Quickstart Installer in the *Installing and Configuring Oracle Identity and Access Management*.

The other option is to install the required product distributions - Infrastructure, Oracle SOA Suite, and Oracle Identity Manager 14c (14.1.2.1.0) separately. To do this, complete the following steps:

1. Sign in to the target system.
2. Download the following from Oracle Technology Network or Oracle Software Delivery Cloud to your target system:
   - If you not yet installed Oracle Fusion Middleware Infrastructure, then download Oracle Fusion Middleware Infrastructure (`fmw_14.1.2.0.0_infrastructure.jar`)
   - Oracle SOA Suite (`fmw_14.1.2.0.0_soa.jar`)

- Oracle Identity and Access Management 14c (`fmw_14.1.2.1.0_idm_Disk1_1of1.zip`, which contains `fmw_14.1.2.1.0_idm.jar`) from OTN or Oracle Software Delivery Cloud.

> **Note:**
>
> Ensure that the *ORACLE_HOME* folder exists and it does not contain any files or folders. If there are any remaining files or folders in the *ORACLE_HOME* folder, delete them.

3. Change to the directory where you downloaded the 14c (14.1.2.1.0) product distribution.
4. If you have already installed Oracle Fusion Middleware Infrastructure (`fmw_14.1.2.0.0_infrastructure.jar`), go to step 15.
5. Start the installation program for Oracle Fusion Middleware Infrastructure pointing to the new JDK. Pointing to the new JDK location helps to skip a step later in the upgrade process.

   Run the following commands:

   - (UNIX) *NEW_JDK_HOME*`/bin/java -jar fmw_14.1.2.0.0_infrastructure.jar`
   - (Windows) *NEW_JDK_HOME*`\bin\java -jar fmw_14.1.2.0.0_infrastructure.jar`

   > **Note:**
   >
   > If the `user_projects` directory and the `domain-registry.xml` file are left in place in the *ORACLE_HOME*, the `-novalidation` flag needs to be used to avoid the install from failing.
   >
   > Following is an example of the failure message:
   >
   > ```
   > Verifying data......
   > [VALIDATION] [ERROR]:INST-07319: Validation of Oracle Home
   > location failed. The location specified already exists and is a
   > nonempty directory and not a valid Oracle Home
   > [VALIDATION] [SUGGESTION]:Provide an empty or nonexistent
   > directory location, or a valid existing Oracle Home
   > installation Failed. Exiting installation due to data validation
   > failure.
   > he Oracle Universal Installer failed. Exiting.
   > ```

6. On UNIX operating systems, the Installation Inventory Setup screen appears if this is the first time you are installing an 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, and click **Next**.

> **Note:**
>
> The Installation Inventory Setup screen does not appear on Windows operating systems.

7. On the Welcome screen, review the information to make sure that you have met all the prerequisites. Click **Next**.
8. On the Auto Updates screen, select an option:
   - **Skip Auto Updates**: If you do not want your system to check for software updates at this time.
   - **Select patches from directory**: To navigate to a local directory if you downloaded patch files.
   - **Search My Oracle Support for Updates**: To automatically download software updates if you have a My Oracle Support account. You must enter Oracle Support credentials then click **Search**. To configure a proxy server for the installer to access My Oracle Support, click **Proxy Settings**. Click **Test Connection** to test the connection.

   Click **Next**.
9. On the Installation Location screen, specify the location for the existing Oracle home directory and click **Next**.

   For more information about Oracle Fusion Middleware directory structure, see Understanding Directories for Installation and Configuration in *Oracle Fusion Middleware Planning an Installation of Oracle Fusion Middleware*.
10. On the Installation Type screen, select **Fusion Middleware Infrastructure**.

    Click **Next**.
11. The Prerequisite Checks screen analyzes the host computer to ensure that the specific operating system prerequisites have been met.

    To view the list of tasks that are 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).
12. On the Installation Summary screen, verify the installation options that 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.
13. On the Installation Progress screen, when the progress bar displays 100%, click **Finish** to dismiss the installer, or click **Next** to see a summary.
14. The Installation Complete screen displays the Installation Location and the Feature Sets that are installed. Review this information and click **Finish** to close the installer.
15. After you have installed Oracle Fusion Middleware Infrastructure, enter the following command to start the installer for your product distribution and repeat the steps above to navigate through the installer screens:

    For installing Oracle SOA Suite 14*c* (14.1.2.0.0), run the following installer:

> **Note:**
>
> On the Installation Type screen, for Oracle SOA Suite, select **Oracle SOA Suite**.

- (UNIX) `NEW_JDK_HOME/bin/java -jar fmw_14.1.2.0.0_soa.jar`
- (Windows) `NEW_JDK_HOME\bin\java -jar fmw_14.1.2.0.0_soa.jar`

For installing Oracle Identity Manager 14c (14.1.2.1.0), run the following installer:

> **Note:**
>
> On the Installation Type screen, for Oracle Identity Manager, select **Collocated Oracle Identity and Access Manager**.

- (UNIX) `NEW_JDK_HOME/bin/java -jar fmw_12.2.1.3.0_idm.jar`
- (Windows) `NEW_JDK_HOME\bin\java -jar fmw_12.2.1.3.0_idm.jar`

For more information about installing Oracle Identity Manager 14c (14.1.2.1.0), see Installing the Oracle Identity and Access Management Software in the *Installing and Configuring Oracle Identity and Access Management*.

Running a Pre-Upgrade Readiness Check

To identify potential issues with the upgrade, Oracle recommends that you run a readiness check before you start the upgrade process. Be aware that the readiness check may not be able to discover all potential issues with your upgrade. An upgrade may still fail, even if the readiness check reports success.

- About Running a Pre-Upgrade Readiness Check
  You can run the Upgrade Assistant in `-readiness` mode to detect issues before you perform the actual upgrade. You can run the readiness check in GUI mode using the Upgrade Assistant or in silent mode using a response file.
- Starting the Upgrade Assistant in Readiness Mode
  Use the `-readiness` parameter to start the Upgrade Assistant in readiness mode.
- Performing a Readiness Check with the Upgrade Assistant
  Navigate through the screens in the Upgrade Assistant to complete the pre-upgrade readiness check.
- Understanding the Readiness Report
  After performing a readiness check for your domain, review the report to determine whether you need to take any action for a successful upgrade.

About Running a Pre-Upgrade Readiness Check

You can run the Upgrade Assistant in `-readiness` mode to detect issues before you perform the actual upgrade. You can run the readiness check in GUI mode using the Upgrade Assistant or in silent mode using a response file.

The Upgrade Assistant readiness check performs a read-only, pre-upgrade review of your Fusion Middleware schemas and WebLogic domain configurations that are at a supported starting point. The review is a read-only operation.

The readiness check generates a formatted, time-stamped readiness report so you can address potential issues before you attempt the actual upgrade. If no issues are detected, you can begin the upgrade process. Oracle recommends that you read this report thoroughly before performing an upgrade.

You can run the readiness check while your existing Oracle Fusion Middleware domain is online (while other users are actively using it) or offline.

You can run the readiness check any number of times before performing any actual upgrade. However, do not run the readiness check after an upgrade has been performed, as the report results may differ from the result of pre-upgrade readiness checks.

> **Note:**
>
> To prevent performance from being affected, Oracle recommends that you run the readiness check during off-peak hours.

Starting the Upgrade Assistant in Readiness Mode

Use the `-readiness` parameter to start the Upgrade Assistant in readiness mode.

To perform a readiness check on your pre-upgrade environment with the Upgrade Assistant:

1. Go to the `oracle_common/upgrade/bin` directory:
   - (UNIX) *`ORACLE_HOME`*`/oracle_common/upgrade/bin`
   - (Windows) *`ORACLE_HOME`*`\oracle_common\upgrade\bin`

   Where, *`ORACLE_HOME`* is the 14c (14.1.2.1.0) Oracle Home.
2. Start the Upgrade Assistant.
   - (UNIX) `./ua -readiness`
   - (Windows) `ua.bat -readiness`

   > **Note:**
   >
   > If the `DISPLAY` environment variable is not set up properly to allow for GUI mode, you may encounter the following error:
   >
   > ```
   > Xlib: connection to ":1.0" refused by server
   > Xlib: No protocol specified
   > ```
   >
   > To resolve this issue you need to set the `DISPLAY` variable to the host and desktop where a valid `X` environment is working.
   >
   > For example, if you are running an `X` environment inside a VNC on the local host in desktop 6, then you would set `DISPLAY=:6`. If you are running `X` on a remote host on desktop 1 then you would set this to `DISPLAY=remoteHost:1`.

   For information about other parameters that you can specify on the command line, see:

- Upgrade Assistant Parameters

Upgrade Assistant Parameters

When you start the Upgrade Assistant from the command line, you can specify additional parameters.

Table 3-2 Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-readiness`	Required for readiness checks **Note**: Readiness checks cannot be performed on standalone installations (those not managed by the WebLogic Server).	Performs the upgrade readiness check without performing an actual upgrade. Schemas and configurations are checked. Do not use this parameter if you have specified the `-examine` parameter.
`-threads`	Optional	Identifies the number of threads available for concurrent schema upgrades or readiness checks of the schemas. The value must be a positive integer in the range 1 to 8. The default is 4.
`-response`	Required for silent upgrades or silent readiness checks	Runs the Upgrade Assistant using inputs saved to a response file generated from the data that is entered when the Upgrade Assistant is run in GUI mode. Using this parameter runs the Upgrade Assistant in *silent mode* (without displaying Upgrade Assistant screens).
`-examine`	Optional	Performs the examine phase but does not perform an actual upgrade. Do not specify this parameter if you have specified the `-readiness` parameter.
`-logLevel` *attribute*	Optional	Sets the logging level, specifying one of the following attributes: • `TRACE` • `NOTIFICATION` • `WARNING` • `ERROR` • `INCIDENT_ERROR` The default logging level is `NOTIFICATION`. Consider setting the `-logLevel TRACE` attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if `-logLevel TRACE` is used.

Table 3-2 (Cont.) Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-logDir` *location*	Optional	Sets the default location of upgrade log files and temporary files. You must specify an existing, writable directory where the Upgrade Assistant creates log files and temporary files. The default locations are: (UNIX) *ORACLE_HOME*`/oracle_common/upgrade/logs` *ORACLE_HOME*`/oracle_common/upgrade/temp` (Windows) *ORACLE_HOME*`\oracle_common\upgrade\logs` *ORACLE_HOME*`\oracle_common\upgrade\temp`
`-help`	Optional	Displays all of the command-line options.

Performing a Readiness Check with the Upgrade Assistant

Navigate through the screens in the Upgrade Assistant to complete the pre-upgrade readiness check.

Readiness checks are performed only on schemas or component configurations that are at a supported upgrade starting point.

To complete the readiness check:

1. On the Welcome screen, review information about the readiness check. Click **Next**.
2. On the Readiness Check Type screen, select **Domain Based**.

   The **Domain Based** option enables the Upgrade Assistant to discover and select all upgrade-eligible schemas or component configurations in the domain specified in the **Domain Directory** field.

   When you select this option, the screen name changes to Schemas and Configuration.

   Leave the default selection if you want the Upgrade Assistant to check all schemas and component configurations at the same time, or select a specific option:

   - **Include checks for all schemas** to discover and review all components that have a schema available to upgrade.
   - **Include checks for all configurations** to review component configurations for a managed WebLogic Server domain.
3. In the **Domain Directory** field, select the 12c (12.2.1.4.0) domain folder that was copied to the 14c (14.1.2.1.0) setup machine. If the 14c (14.1.2.1.0) setup is on the same machine as the 12c Release, provide the 12c domain home location during the readiness check.

   Click **Next**.

4. The Component List screen displays the list of components whose schema will be upgraded.

   Click **Next**.

5. On the Schema Credentials screen, specify the database credentials to connect to the selected 12c (12.2.1.4.0) schema: **Database Type**, **DBA User Name**, and **DBA Password**. As part of the pre-upgrade requirements, you had created the required user, see Creating a Non-SYSDBA User to Run the Upgrade Assistant.

   Then click **Connect**.

   > **Note:**
   >
   > Oracle database is the default database type. Make sure that you select the correct database type before you continue. If you discover that you selected the wrong database type, do not go back to this screen to change it to the correct type. Instead, close the Upgrade Assistant and restart the readiness check with the correct database type selected to ensure that the correct database type is applied to all schemas.

   Select the **Schema User Name** option and specify the **Schema Password**.

   > **Note:**
   >
   > The Upgrade Assistant automatically enables the default credentials. If you are unable to connect, ensure that you manually enter the credentials for your schema before you continue.

   Click **Next** until all schema connections are validated (the screen name changes based on the schema selected).

   > **Note:**
   >
   > If you encounter any connection failure, check the cause and fix it.

6. On the Readiness Summary screen, review the summary of the readiness checks that will be performed based on your selections.

   If you want to save your selections to a response file to run the Upgrade Assistant again later in response (or silent) mode, click **Save Response File** and provide the location and name of the response file. A silent upgrade performs exactly the same function that the Upgrade Assistant performs, but you do not have to manually enter the data again.

   For a detailed report, click **View Log**.

   Click **Next**.

7. On the Readiness Check screen, review the status of the readiness check. The process can take several minutes.

   If you are checking multiple components, the progress of each component displays in its own progress bar in parallel.

   When the readiness check is complete, click **Continue**.

The following components are marked as **ready for upgrade** although they are not upgraded. Ignore the **ready for upgrade** message against these components:

- Oracle JRF
- Common Infrastructure Services
- Oracle Web Services Manager

8. On the End of Readiness screen, review the results of the readiness check (**Readiness Success** or **Readiness Failure**):
   - If the readiness check is successful, click **View Readiness Report** to review the complete report. Oracle recommends that you review the Readiness Report before you perform the actual upgrade even when the readiness check is successful. Use the **Find** option to search for a particular word or phrase within the report. The report also indicates where the completed Readiness Check Report file is located.
   - If the readiness check encounters an issue or error, click **View Log** to review the log file, identify and correct the issues, and then restart the readiness check. The log file is managed by the command-line options you set.

Understanding the Readiness Report

After performing a readiness check for your domain, review the report to determine whether you need to take any action for a successful upgrade.

The format of the readiness report file is:

`readiness_`*`timestamp`*`.txt`

where *`timestamp`* indicates the date and time of when the readiness check was run.

A readiness report contains the following information:

Table 3-3 Readiness Report Elements

Report Information	Description	Required Action
Overall Readiness Status: SUCCESS or FAILURE	The top of the report indicates whether the readiness check passed or completed with one or more errors.	If the report completed with one or more errors, search for FAIL and correct the failing issues before attempting to upgrade. You can re-run the readiness check as many times as necessary before an upgrade.
Timestamp	The date and time that the report was generated.	No action required.
Log file location *`ORACLE_HOME`*`/oracle_common/upgrade/logs`	The directory location of the generated log file.	No action required.
Readiness report location *`ORACLE_HOME`*`/oracle_common/upgrade/logs`	The directory location of the generated readiness report.	No action required.
Names of components that were checked	The names and versions of the components included in the check and status.	If your domain includes components that cannot be upgraded to this release, such as SOA Core Extension, do not attempt an upgrade.

Table 3-3 (Cont.) Readiness Report Elements

Report Information	Description	Required Action
Names of schemas that were checked	The names and current versions of the schemas included in the check and status.	Review the version numbers of your schemas. If your domain includes schemas that cannot be upgraded to this release, do not attempt an upgrade.
Individual Object Test Status: FAIL	The readiness check test detected an issue with a specific object.	Do not upgrade until all failed issues have been resolved.
Individual Object Test Status: PASS	The readiness check test detected no issues for the specific object.	If your readiness check report shows only the PASS status, you can upgrade your environment. Note, however, that the Readiness Check cannot detect issues with externals such as hardware or connectivity during an upgrade. You should always monitor the progress of your upgrade.
Completed Readiness Check of <Object> Status: FAILURE	The readiness check detected one or more errors that must be resolved for a particular object such as a schema, an index, or datatype.	Do not upgrade until all failed issues have been resolved.
Completed Readiness Check of <Object> Status: SUCCESS	The readiness check test detected no issues.	No action required.

Here is a sample Readiness Report file. Your report may not include all of these checks.

```
This readiness check report was created on Wed Dec 02 05:47:33 PST 2020 Log
file is located at:
/oracle/work/middleware_latest/oracle_common/upgrade/logs/
ua2020-12-02-05-35-03AM.log
Readiness Check Report File:
/oracle/work/middleware_latest/oracle_common/upgrade/logs/
readiness2020-12-02-05-47-33AM.txt
Domain Directory:
/oracle/work/middleware_1212/user_projects/domains/oim_domain

Starting readiness check of components.

Oracle Platform Security Services
    Starting readiness check of Oracle Platform Security Services.
      Schema User Name: DEV_OPSS
      Database Type: Oracle Database
      Database Connect String: example.oracle.com:1521:oimdb
      VERSION Schema DEV_OPSS is currently at version 11.1.1.9.0.
Readiness checks will now be performed.
    Starting schema test:  TEST_DATABASE_VERSION  Test that the database
server version number is supported for upgrade
      INFO   Database product version: Oracle Database 11g Enterprise Edition
Release 11.2.0.4.0 - 64bit Production With the Partitioning, OLAP, Data
Mining and Real Application Testing options
    Completed schema test: TEST_DATABASE_VERSION --> Test that the database
server version number is supported for upgrade +++ PASS
    Starting schema test:  TEST_REQUIRED_TABLES  Test that the schema
contains all the required tables
```

```
    Completed schema test: TEST_REQUIRED_TABLES --> Test that the schema
contains all the required tables +++ PASS
    Starting schema test:  Test that the schema does not contain any
unexpected tables  TEST_UNEXPECTED_TABLES
    Completed schema test: Test that the schema does not contain any
unexpected tables --> TEST_UNEXPECTED_TABLES +++ Test that the schema does
not contain any unexpected tables
    Starting schema test:  TEST_ENOUGH_TABLESPACE  Test that the schema
tablespaces automatically extend if full
    Completed schema test: TEST_ENOUGH_TABLESPACE --> Test that the schema
tablespaces automatically extend if full +++ PASS
    Starting schema test:  TEST_USER_TABLESPACE_QUOTA  Test that tablespace
quota for this user is sufficient to perform the upgrade
    Completed schema test: TEST_USER_TABLESPACE_QUOTA --> Test that
tablespace quota for this user is sufficient to perform the upgrade +++ PASS
    Starting schema test:  TEST_ONLINE_TABLESPACE  Test that schema
tablespaces are online
    Completed schema test: TEST_ONLINE_TABLESPACE --> Test that schema
tablespaces are online +++ PASS
    Starting permissions test:  TEST_DBA_TABLE_GRANTS  Test that DBA user has
privilege to view all user tables
    Completed permissions test: TEST_DBA_TABLE_GRANTS --> Test that DBA user
has privilege to view all user tables +++ PASS
    Starting schema test:  TEST_MISSING_COLUMNS  Test that tables and views
are not missing any required columns
    Completed schema test: TEST_MISSING_COLUMNS --> Test that tables and
views are not missing any required columns +++ PASS
    Starting schema test:  TEST_UNEXPECTED_COLUMNS  Test that tables and
views do not contain any unexpected columns
    Completed schema test: TEST_UNEXPECTED_COLUMNS --> Test that tables and
views do not contain any unexpected columns +++ PASS
    Starting datatype test for table CT_29: TEST_COLUMN_DATATYPES_V2 --> Test
that all table columns have the proper datatypes
    Completed datatype test for table CT_29: TEST_COLUMN_DATATYPES_V2
--> Test that all table columns have the proper datatypes +++ PASS
    Starting index test for table JPS_ENTITY_LOCK: TEST_REQUIRED_INDEXES
--> Test that the table contains all the required indexes
    Completed index test for table JPS_ENTITY_LOCK:
TEST_REQUIRED_INDEXES --> Test that the table contains all the required
indexes +++ PASS
   Starting index test for table CT_9_3:  TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes
    Completed index test for table CT_9_3: TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes +++ PASS
    Starting schema test:  UPGRADE_SCRIPT_TEST  Test that the middleware
contains the required Oracle Platform Security Services upgrade script
    Completed schema test: UPGRADE_SCRIPT_TEST --> Test that the middleware
contains the required Oracle Platform Security Services upgrade script +++
PASS
    Starting schema test:  PRIVILEGES_TEST  Test that the Oracle Platform
Security Services schema has appropriate system privileges
    Completed schema test: PRIVILEGES_TEST --> Test that the Oracle Platform
Security Services schema has appropriate system privileges +++ PASS
    Starting schema test:  SEQUENCE_TEST  Test that the Oracle Platform
Security Services schema sequence and its properties are valid
    Completed schema test: SEQUENCE_TEST --> Test that the Oracle Platform
```

```
Security Services schema sequence and its properties are valid
+++ PASS
    Finished readiness check of Oracle Platform Security Services with
status: SUCCESS.

Oracle Metadata Services
    Starting readiness check of Oracle Metadata Services.
      Schema User Name: DEV_MDS
      Database Type: Oracle Database
      Database Connect String: example.oracle.com:1521:oimdb
      VERSION Schema DEV_MDS is currently at version 11.1.1.9.0.
Readiness checks will now be performed.
    Starting schema test:  TEST_REQUIRED_TABLES  Test that the schema
contains all the required tables
    Completed schema test: TEST_REQUIRED_TABLES --> Test that the schema
contains all the required tables +++ PASS
    Starting schema test:  TEST_REQUIRED_PROCEDURES  Test that the schema
contains all the required stored procedures
    Completed schema test: TEST_REQUIRED_PROCEDURES --> Test that the schema
contains all the required stored procedures +++ PASS
    Starting schema test:  TEST_REQUIRED_VIEWS  Test that the schema contains
all the required database views
    Completed schema test: TEST_REQUIRED_VIEWS --> Test that the schema
contains all the required database views +++ PASS
    Starting index test for table MDS_ATTRIBUTES: TEST_REQUIRED_INDEXES
--> Test that the table contains all the required indexes
    Starting schema test:  TEST_USER_TABLESPACE_QUOTA  Test that tablespace
quota for this user is sufficient to perform the upgrade
    Completed schema test: TEST_USER_TABLESPACE_QUOTA --> Test that
tablespace quota for this user is sufficient to perform the upgrade +++ PASS
    Starting schema test:  TEST_ONLINE_TABLESPACE  Test that schema
tablespaces are online
    Completed schema test: TEST_ONLINE_TABLESPACE --> Test that schema
tablespaces are online +++ PASS
    Starting schema test:  TEST_DATABASE_VERSION  Test that the database
server version number is supported for upgrade
      INFO   Database product version: Oracle Database 11g Enterprise Edition
Release 11.2.0.4.0 - 64bit Production With the Partitioning, OLAP, Data
Mining and Real Application Testing options
    Completed schema test: TEST_DATABASE_VERSION --> Test that the database
server version number is supported for upgrade +++ PASS
    Finished readiness check of Oracle Metadata Services with status:
SUCCESS.

User Messaging Service
    Starting readiness check of User Messaging Service.
      Schema User Name: DEV_ORASDPM
      Database Type: Oracle Database
      Database Connect String: example.oracle.com:1521:oimdb
      VERSION Schema DEV_ORASDPM is currently at version 11.1.1.9.0.
Readiness checks will now be performed.
    Starting schema test:  TEST_DATABASE_VERSION  Test that the database
server version number is supported for upgrade
      INFO   Database product version: Oracle Database 11g Enterprise Edition
Release 11.2.0.4.0 - 64bit Production With the Partitioning, OLAP, Data
Mining and Real Application Testing options
```

```
    Completed schema test: TEST_DATABASE_VERSION --> Test that the database
server version number is supported for upgrade +++ PASS
    Starting column test for table RULE_SET:
TEST_UNEXPECTED_TABLE_COLUMNS --> Test that the table does not contain any
unexpected columns
    Completed column test for table RULE_SET:
TEST_UNEXPECTED_TABLE_COLUMNS --> Test that the table does not contain any
unexpected columns +++ PASS
    Starting column test for table STATUS: TEST_UNEXPECTED_TABLE_COLUMNS
--> Test that the table does not contain any unexpected columns
    Completed column test for table STATUS:
TEST_UNEXPECTED_TABLE_COLUMNS --> Test that the table does not contain any
unexpected columns +++ PASS
    Starting column test for table STATUS_ORPHAN:
TEST_UNEXPECTED_TABLE_COLUMNS --> Test that the table does not contain any
unexpected columns
    Completed column test for table STATUS_ORPHAN:
TEST_UNEXPECTED_TABLE_COLUMNS --> Test that the table does not contain any
unexpected columns +++ PASS
    Starting column test for table USER_DEVICE:
TEST_UNEXPECTED_TABLE_COLUMNS --> Test that the table does not contain any
unexpected columns
    Completed column test for table USER_DEVICE:
TEST_UNEXPECTED_TABLE_COLUMNS --> Test that the table does not contain any
unexpected columns +++ PASS
    Finished readiness check of User Messaging Service with status: SUCCESS.

Oracle SOA
    Starting readiness check of Oracle SOA.
      Schema User Name: DEV_SOAINFRA
      Database Type: Oracle Database
      Database Connect String: example.oracle.com:1521:oimdb
      VERSION Schema DEV_SOAINFRA is currently at version 11.1.1.9.0.
Readiness checks will now be performed.
    Starting schema test:  TEST_DATABASE_VERSION  Test that the database
server version number is supported for upgrade
      INFO   Database product version: Oracle Database 11g Enterprise Edition
Release 11.2.0.4.0 - 64bit Production With the Partitioning, OLAP, Data
Mining and Real Application Testing options
    Completed schema test: TEST_DATABASE_VERSION --> Test that the database
server version number is supported for upgrade +++ PASS
    Starting schema test:  TEST_REQUIRED_TABLES  Test that the schema
contains all the required tables
    Completed schema test: TEST_REQUIRED_TABLES --> Test that the schema
contains all the required tables +++ PASS
    Starting schema test:  TEST_REQUIRED_PROCEDURES  Test that the schema
contains all the required stored procedures
    Completed schema test: TEST_REQUIRED_PROCEDURES --> Test that the schema
contains all the required stored procedures +++ PASS
    Starting schema test:  TEST_REQUIRED_VIEWS  Test that the schema contains
all the required database views
    Completed schema test: TEST_REQUIRED_VIEWS --> Test that the schema
contains all the required database views +++ PASS
    Starting schema test:  TEST_ENOUGH_TABLESPACE  Test that the schema
tablespaces automatically extend if full
    Completed schema test: TEST_ENOUGH_TABLESPACE --> Test that the schema
```

```
tablespaces automatically extend if full +++ PASS
    Starting schema test:  TEST_ONLINE_TABLESPACE  Test that schema
tablespaces are online
    Completed schema test: TEST_ONLINE_TABLESPACE --> Test that schema
tablespaces are online +++ PASS
    Starting schema test:  TEST_USER_TABLESPACE_QUOTA  Test that tablespace
quota for this user is sufficient to perform the upgrade
    Completed schema test: TEST_USER_TABLESPACE_QUOTA --> Test that
tablespace quota for this user is sufficient to perform the upgrade +++ PASS
    Starting schema test:  SOA_TABLESPACE_VALIDATION  Test SOAINFRA schema
for enough default table space and temp table space.
    Completed schema test: SOA_TABLESPACE_VALIDATION --> Test SOAINFRA schema
for enough default table space and temp table space. +++ PASS
    Starting schema test:  SOA_INSTANCE_VALIDATION  Test SOAINFRA schema for
inconsistencies of instance data.
    Completed schema test: SOA_INSTANCE_VALIDATION --> Test SOAINFRA schema
for inconsistencies of instance data. +++ PASS
    Finished readiness check of Oracle SOA with status: SUCCESS.

Oracle Identity Manager
    Starting readiness check of Oracle Identity Manager.
      Schema User Name: DEV_OIM
      Database Type: Oracle Database
      Database Connect String: example.oracle.com:1521:oimdb
    Starting schema test:  examine  Calling examine method
      INFO Examine is successful
    Completed schema test: Examine --> Testing schema version +++ PASS
    Starting schema test:  TEST_MDS_BACKUP  Taking backup of MDS data related
to OIM to handle any unseen situation during upgrade.
      INFO MDSBackup passes. Backup of MDS data related to OIM is here:
/oracle/work/middleware_latest/oracle_common/upgrade/temp/mdsBackup/
    Completed schema test: TEST_MDS_BACKUP --> Taking backup of MDS data
related to OIM to handle any unseen situration during upgrade. +++ PASS
    Finished readiness check of Oracle Identity Manager with status:
SUCCESS.

User Messaging Service
    Starting readiness check of User Messaging Service.
    Starting config test:  TEST_USERMESSAGINGCONFIG  Test that configuration
file usermessagingconfig.xml is accessible, in place and valid.
    Completed config test: TEST_USERMESSAGINGCONFIG --> Configuration file
usermessagingconfig.xml is accessible, in place and valid. +++ PASS
    Starting config test:  TEST_ALREADY_UPGRADED  Test that configuration is
not already upgraded.
    Completed config test: TEST_ALREADY_UPGRADED --> Configuration is not
already upgraded. +++ PASS
    Finished readiness check of User Messaging Service with status: SUCCESS.

Oracle Identity Manager
    Starting readiness check of Oracle Identity Manager.
      INFO There are no configuration readiness tests for Oracle Identity
Manager.
    Finished readiness check of Oracle Identity Manager with status:
SUCCESS.

Oracle JRF
```

```
    Starting readiness check of Oracle JRF.
    Finished readiness check of Oracle JRF with status: SUCCESS.

System Components Infrastructure
    Starting readiness check of System Components Infrastructure.
    Starting config test:  TEST_SOURCE_CONFIG  Checking the source
configuration.
      INFO
/oracle/work/middleware_1212/user_projects/oim_domain/opmn/topology.xml
was not found. No upgrade is needed.
    Completed config test: TEST_SOURCE_CONFIG --> Checking the source
configuration. +++ PASS
    Finished readiness check of System Components Infrastructure with
status: ALREADY_UPGRADED.

Common Infrastructure Services
    Starting readiness check of Common Infrastructure Services.
    Starting config test:  CIEConfigPlugin.readiness.test  This tests the
readiness of the domain from CIE side.
    Completed config test: CIEConfigPlugin.readiness.test --> This tests the
readiness of the domain from CIE side. +++ PASS
    Finished readiness check of Common Infrastructure Services with
status: SUCCESS.

Oracle Web Services Manager
    Starting readiness check of Oracle Web Services Manager.
    Completed config test: BOOTSTRAP_PROPERTIES_CHECK --> Bootstrap
properties check +++ PASS
    Completed config test: CONFIGURATION_PROPERTIES_CHECK --> Configuration
properties check +++ PASS
    Completed config test: TOKEN_TRUST_PROPERTIES_CHECK --> Trust issuer
properties check +++ PASS
    Completed config test: MDS_REPOSITORY_CONNECTIVITY_CHECK --> MDS
repository connectivity check +++ PASS
    Finished readiness check of Oracle Web Services Manager with status:
SUCCESS.

Finished readiness check of components.
```

Note:

You can ignore the missing index error in the readiness report. This is a known issue. The corresponding missing index is added during the schema upgrade operation. This error does not occur if the schema to be upgraded was created in 12*c* using the RCU.

Upgrading Product Schemas

After stopping servers and processes, use the Upgrade Assistant to upgrade supported product schemas to the current release of Oracle Fusion Middleware.

The Upgrade Assistant allows you to upgrade individually selected schemas or all schemas associated with a domain. The option you select determines which Upgrade Assistant screens you will use.

> **Note:**
>
> High waits and performance degradation may be seen due to 'library cache lock' (cycle)<='library cache lock' for DataPump Worker (DW) processes in the 12.2 RAC environment. To resolve this issue, you should disable S-Optimization by using the following command:
>
> ```
> ALTER SYSTEM SET "_lm_share_lock_opt"=FALSE SCOPE=SPFILE SID='*';
> ```
>
> After running the above command, restart all the RAC instances. After the upgrade is complete, you can reset the parameter by using the following command:
>
> ```
> alter system reset "_lm_share_lock_opt" scope=spfile sid='*';
> ```

- Identifying Existing Schemas Available for Upgrade
  This optional step can be used before an upgrade to query the schema version registry table. This table contains schema information such as the schema owner, version number, component name and ID, date of creation and modification, and custom prefixes.
- Starting the Upgrade Assistant
  Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 14c (14.1.2.1.0).
- Upgrading Oracle Identity Manager Schemas Using the Upgrade Assistant
  Navigate through the screens in the Upgrade Assistant to upgrade the product schemas.
- Verifying the Schema Upgrade
  After completing all the upgrade steps, verify that the upgrade was successful by checking that the schema version in `schema_version_registry` has been properly updated.

Identifying Existing Schemas Available for Upgrade

This optional step can be used before an upgrade to query the schema version registry table. This table contains schema information such as the schema owner, version number, component name and ID, date of creation and modification, and custom prefixes.

You can let the Upgrade Assistant upgrade all of the schemas in the domain, or you can select individual schemas to upgrade. To help decide, follow these steps to view a list of all the schemas that are available for an upgrade:

1. If you are using an Oracle database, connect to the database by using an account that has Oracle DBA privileges, and run the following from SQL*Plus:

```
SET LINE 120
COLUMN MRC_NAME FORMAT A14
COLUMN COMP_ID FORMAT A20
COLUMN VERSION FORMAT A12
COLUMN STATUS FORMAT A9
COLUMN UPGRADED FORMAT A8
SELECT MRC_NAME, COMP_ID, OWNER, VERSION, STATUS, UPGRADED FROM
SCHEMA_VERSION_REGISTRY WHERE OWNER LIKE UPPER('<PREFIX>_%');
```

2. Examine the report that is generated.

Notes:

- After the upgrade you can generate the report again to see the updated versions of your schemas. If an upgrade was not needed for a schema, the `schema_version_registry` table retains the schema at its pre-upgrade version.
- If your existing schemas are not from a supported version, then you must upgrade them to a supported version before using the 14c (14.1.2.1.0) upgrade procedures. Refer to your pre-upgrade version documentation for more information.
- If you used an OID-based policy store in the earlier versions, make sure to create a new OPSS schema before you perform the upgrade. After the upgrade, the OPSS schema remains an LDAP-based store.
- You can only upgrade schemas for products that are available for upgrade in Oracle Fusion Middleware release 14c (14.1.2.1.0). Do not attempt to upgrade a domain that includes components that are not yet available for upgrade to 14c (14.1.2.1.0).

Starting the Upgrade Assistant

Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 14c (14.1.2.1.0).

To start the Upgrade Assistant:

> **Note:**
>
> Before you start the Upgrade Assistant, make sure that the JVM character encoding is set to UTF-8 for the platform on which the Upgrade Assistant is running. If the character encoding is not set to UTF-8, then you will not be able to download files containing Unicode characters in their names. This can cause the upgrade to fail. To set the character encoding, run the following:
>
> UNIX operating systems:
>
> ```
> export UA_PROPERTIES="-Dfile.encoding=UTF-8 ${UA_PROPERTIES}"
> ```
>
> Windows operating systems:
>
> ```
> set UA_PROPERTIES=-Dfile.encoding=UTF-8 %UA_PROPERTIES%
> ```

1. Go to the `oracle_common/upgrade/bin` directory:
   - (UNIX) *ORACLE_HOME*`/oracle_common/upgrade/bin`
   - (Windows) *ORACLE_HOME*`\oracle_common\upgrade\bin`
2. Start the Upgrade Assistant:
   - (UNIX) `./ua`
   - (Windows) `ua.bat`

For information about other parameters that you can specify on the command line, such as logging parameters, see:

- Upgrade Assistant Parameters

Upgrade Assistant Parameters

When you start the Upgrade Assistant from the command line, you can specify additional parameters.

Table 3-4 Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-readiness`	Required for readiness checks **Note**: Readiness checks cannot be performed on standalone installations (those not managed by the WebLogic Server).	Performs the upgrade readiness check without performing an actual upgrade. Schemas and configurations are checked. Do not use this parameter if you have specified the `-examine` parameter.
`-threads`	Optional	Identifies the number of threads available for concurrent schema upgrades or readiness checks of the schemas. The value must be a positive integer in the range 1 to 8. The default is 4.

Table 3-4 (Cont.) Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-response`	Required for silent upgrades or silent readiness checks	Runs the Upgrade Assistant using inputs saved to a response file generated from the data that is entered when the Upgrade Assistant is run in GUI mode. Using this parameter runs the Upgrade Assistant in *silent mode* (without displaying Upgrade Assistant screens).
`-examine`	Optional	Performs the examine phase but does not perform an actual upgrade. Do not specify this parameter if you have specified the `-readiness` parameter.
`-logLevel` *attribute*	Optional	Sets the logging level, specifying one of the following attributes: • `TRACE` • `NOTIFICATION` • `WARNING` • `ERROR` • `INCIDENT_ERROR` The default logging level is `NOTIFICATION`. Consider setting the `-logLevel TRACE` attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if `-logLevel TRACE` is used.
`-logDir` *location*	Optional	Sets the default location of upgrade log files and temporary files. You must specify an existing, writable directory where the Upgrade Assistant creates log files and temporary files. The default locations are: (UNIX) *ORACLE_HOME*`/oracle_common/upgrade/logs` *ORACLE_HOME*`/oracle_common/upgrade/temp` (Windows) *ORACLE_HOME*`\oracle_common\upgrade\logs` *ORACLE_HOME*`\oracle_common\upgrade\temp`
`-help`	Optional	Displays all of the command-line options.

Upgrading Oracle Identity Manager Schemas Using the Upgrade Assistant

Navigate through the screens in the Upgrade Assistant to upgrade the product schemas.

To upgrade product schemas with the Upgrade Assistant:

1. On the Welcome screen, review an introduction to the Upgrade Assistant and information about important pre-upgrade tasks. Click **Next**.

   > **Note:**
   >
   > For more information about any Upgrade Assistant screen, click **Help** on the screen.

2. On the Upgrade Type screen, select the schema upgrade operation that you want to perform:

   - **Individually Selected Schemas** if you want to select individual schemas for upgrade and you do not want to upgrade all of the schemas used by the domain.

     > **Caution:**
     >
     > Upgrade only those schemas that are used to support your 14c (14.1.2.1.0) components. Do not upgrade schemas that are currently being used to support components that are not included in Oracle Fusion Middleware 14c (14.1.2.1.0).

   - **All Schemas Used by a Domain** to allow the Upgrade Assistant to discover and select all components that have a schema available to upgrade in the domain specified in the **Domain Directory** field. This is also known as a *domain assisted schema upgrade*. Additionally, the Upgrade Assistant pre-populates connection information on the schema input screens.

     > **Note:**
     >
     > Oracle recommends that you select **All Schemas Used by a Domain** for most upgrades to ensure all of the required schemas are included in the upgrade.

   > **Note:**
   >
   > If you are upgrading SSL enabled Oracle Identity Manager setup, select **Individually Selected Schemas** option, and then select Oracle Identity Manager schema only. This automatically selects the dependant schemas. For upgrading SSL enabled setup, you must provide the non-SSL Database connection details on the Schema Credentials screen.

3. If you selected **Individually Selected Schemas**: On the Available Components screen, select the components for which you want to upgrade schemas. When you select a component, the schemas and any dependencies are automatically selected.

> **Note:**
>
> - For the individual schema option, the domain configuration is not accessed, and therefore password values are carried forward from the previous screen. If you encounter any connection failure, check the cause and fix it.
> - For the Upgrade Assistant utility to use the correct UMS schema, manually edit the UMS schema by adding `_UMS` as a suffix. For example, edit `DEV` to `DEV_UMS` for successful SOA upgrade.

4. On the Screen name, select the domain folder.

   Click **Next**.

5. On the Component List screen, it will display the list of components whose schema will be upgraded.

   Click **Next**.

6. On the Prerequisites screen, acknowledge that the prerequisites have been met by selecting all the check boxes. Click **Next**.

> **Note:**
>
> The Upgrade Assistant does not verify whether the prerequisites have been met.

7. On the Schema Credentials screen(s), specify the database connection details for each schema you are upgrading (the screen name changes based on the schema selected):
   - Select the database type from the **Database Type** drop-down menu.
   - Enter the database connection details, and click **Connect**.
   - Select the schema you want to upgrade from the **Schema User Name** drop-down menu, and then enter the password for the schema. Be sure to use the correct schema prefix for the schemas you are upgrading.

   Click **Next**.

8. On the Examine screen, review the status of the Upgrade Assistant as it examines each schema, verifying that the schema is ready for upgrade. If the status is **Examine finished**, click **Next**.

   If the examine phase fails, Oracle recommends that you cancel the upgrade by clicking **No** in the Examination Failure dialog. Click **View Log** to see what caused the error and refer to Troubleshooting Your Upgrade in *Upgrading with the Upgrade Assistant* for information on resolving common upgrade errors.

> **Note:**
>
> - If you resolve any issues detected during the examine phase without proceeding with the upgrade, you can start the Upgrade Assistant again without restoring from backup. However, if you proceed by clicking **Yes** in the Examination Failure dialog box, you need to restore your pre-upgrade environment from backup before starting the Upgrade Assistant again.
> - Canceling the examination process has no effect on the schemas or configuration data; the only consequence is that the information the Upgrade Assistant has collected must be collected again in a future upgrade session.

9. On the Upgrade Summary screen, review the summary of the options you have selected for schema upgrade.

   Verify that the correct Source and Target Versions are listed for each schema you intend to upgrade.

   If you want to save these options to a response file to run the Upgrade Assistant again later in response (or silent) mode, click **Save Response File** and provide the location and name of the response file. A silent upgrade performs exactly the same function that the Upgrade Assistant performs, but you do not have to manually enter the data again.

   Click **Upgrade** to start the upgrade process.

10. On the Upgrade Progress screen, monitor the status of the upgrade.

> **Caution:**
>
> Allow the Upgrade Assistant enough time to perform the upgrade. Do not cancel the upgrade operation unless absolutely necessary. Doing so may result in an unstable environment.

   If any schemas are not upgraded successfully, refer to the Upgrade Assistant log files for more information.

> **Note:**
>
> The progress bar on this screen displays the progress of the current upgrade procedure. It does not indicate the time remaining for the upgrade.

   Click **Next**.

11. After the upgrade completes successfully, the Upgrade Assistant provides the upgrade status and lists the next steps to take in the upgrade process. You should review the Upgrade Success screen of the Upgrade Assistant to determine the next steps based on the information provided. The wizard shows the following information:

```
Upgrade Succeeded.

Log File: /u01/oracle/products/12c/identity/oracle_common/upgrade/logs/
ua2020-09-15-18-27-29PM.txt
Post Upgrade Text file: /u01/oracle/products/12c/identity/oracle_common/upgrade/logs/
postupgrade2020-09-15-18-27-29PM.txt
```

```
Next Steps

Oracle SOA
1. The Upgrade Assistant has successfully upgraded all active instances. You can now
close the Upgrade Assistant.
2. The automated upgrade of closed instances will continue in the background after
the Upgrade Assistant is exited and until the SOA server is started,at which point
the upgrade will stop. You can schedule the upgrade of any remaining closed
instances for a time when the SOA server is less busy.
   Close the Upgrade Assistant and use the instance data administration scripts to
administer and monitor the overall progress of this automated upgrade. For more
information see "Administering and Monitoring the Upgrade of SOA Instance Data" in
Upgrading SOA Suite and Business Process Management.
```

Click **Close** to complete the upgrade and close the wizard.

If the upgrade fails: On the Upgrade Failure screen, click **View Log** to view and troubleshoot the errors. The logs are available at *ORACLE_HOME*/oracle_common/upgrade/logs.

> **Note:**
>
> If the upgrade fails, you must restore your pre-upgrade environment from backup, fix the issues, then restart the Upgrade Assistant.

Verifying the Schema Upgrade

After completing all the upgrade steps, verify that the upgrade was successful by checking that the schema version in `schema_version_registry` has been properly updated.

If you are using an Oracle database, connect to the database as a user having Oracle DBA privileges, and run the following from SQL*Plus to get the current version numbers. Be sure to replace <*PREFIX*> with your schema prefix.

```
SET LINE 120
COLUMN MRC_NAME FORMAT A14
COLUMN COMP_ID FORMAT A20
COLUMN VERSION FORMAT A12
COLUMN STATUS FORMAT A9
COLUMN UPGRADED FORMAT A8
SELECT MRC_NAME, COMP_ID, OWNER, EDITION NAME, VERSION, STATUS, UPGRADED FROM
SCHEMA_VERSION_REGISTRY where owner like '<PREFIX>_%';
```

In the query result:

- Verify that the `EDITION NAME` column appears as `ORA$BASE`.
- Check that the number in the `VERSION` column matches the latest version number for that schema. For example, verify that the schema version number is 14.1.2.1.0.

> **Note:**
>
> Not all schema versions will be updated. Some schemas do not require an upgrade to this release and will retain their pre-upgrade version number.

- The `STATUS` field will be either `UPGRADING` or `UPGRADED` during the schema patching operation, and will become `VALID` when the operation is completed.
- If the status appears as `INVALID`, the schema update failed. You should examine the logs files to determine the reason for the failure.
- Synonym objects owned by `IAU_APPEND` and `IAU_VIEWER` will appear as `INVALID`, but that does not indicate a failure.

  They become invalid because the target object changes after the creation of the synonym. The synonyms objects will become valid when they are accessed. You can safely ignore these `INVALID` objects.

Reconfiguring the Domain

Run the Reconfiguration Wizard to reconfigure your domain component configurations to 14c (14.1.2.1.0).

When you reconfigure a WebLogic Server domain, the following items are automatically updated, depending on the applications in the domain:

- WebLogic Server core infrastructure
- Domain version

> **Note:**
>
> Before you begin the domain reconfiguration, note the following limitations:
>
> - Make sure that the original Middleware home does not include any deployments that can cause an error.
> - The Reconfiguration Wizard does not update any of your own applications that are included in the domain.
> - Transforming a non-dynamic cluster domain to a dynamic cluster domain during the upgrade process is not supported.
>
> The dynamic cluster feature is available when running the Reconfiguration Wizard, but Oracle only supports upgrading a non-dynamic cluster upgrade and then adding dynamic clusters. You cannot add dynamic cluster during the upgrade process.

Specifically, when you reconfigure a domain, the following occurs:

- The domain version number in the `config.xml` file for the domain is updated to the Administration Server's installed WebLogic Server version.
- Reconfiguration templates for all installed Oracle products are automatically selected and applied to the domain. These templates define any reconfiguration tasks that are required to make the WebLogic domain compatible with the current WebLogic Server version.
- Start scripts are updated.

  If you want to preserve your modified start scripts, be sure to back them up before starting the Reconfiguration Wizard.

> **Note:**
>
> When the domain reconfiguration process starts, you can't undo the changes that it makes. Before running the Reconfiguration Wizard, ensure that you have backed up the domain as covered in the pre-upgrade checklist. If an error or other interruption occurs while running the Reconfiguration Wizard, you must restore the domain by copying the files and directories from the backup location to the original domain directory. This is the only way to ensure that the domain has been returned to its original state before reconfiguration.

- Backing Up the Domain
- Starting the Reconfiguration Wizard
- Reconfiguring the Oracle Identity Manager Domain
  Navigate through the screens in the Reconfiguration Wizard to reconfigure your existing domain.

Backing Up the Domain

Before running the Reconfiguration Wizard, create a backup copy of the domain directory.

1. Create a backup of the domain directory.
2. Before updating the domain on each remote Managed Server, create a backup copy of the domain directory on each remote machine.
3. Verify that the backed up versions of the domain are complete.

If domain reconfiguration fails for any reason, you must copy all files and directories from the backup directory into the original domain directory to ensure that the domain is returned entirely to its original state before reconfiguration.

Starting the Reconfiguration Wizard

> **Note:**
>
> - Shut down the administration server and all managed servers before starting the reconfiguration process. See Stopping Servers and Processes.
> - If the source is a clustered environment, run the Reconfiguration Wizard on the primary node only, where, primary node is the Administration Server. Use the `Pack`/`Unpack` utility to apply the changes to other cluster members in the domain.

To start the Reconfiguration Wizard in graphical mode:

1. Open the command shell (on UNIX operating systems) or open a command prompt window (on Windows operating systems).
2. Set the following environment variables:
   - `WLS_ALTERNATIVE_TYPES_DIR` - Use the following command:

(Non-Bash): `setenv WLS_ALTERNATIVE_TYPES_DIR ORACLE_HOME/idm/server/loginmodule/wls`

(Bash):`export WLS_ALTERNATIVE_TYPES_DIR=ORACLE_HOME/idm/server/loginmodule/wls`

Where, *ORACLE_HOME* is the 12c Oracle Home.

- `CONFIG_JVM_ARGS` - The `./reconfig.sh` command may display the following error to indicate that the default cache directory is not valid:

```
*sys-package-mgr*: can't create package cache dir
```

To avoid the error, change the cache directory by setting `CONFIG_JVM_ARGS`.

For example: `CONFIG_JVM_ARGS=-Dpython.cachedir=any_writable_directory`.

3. Go to the `oracle_common/common/bin` directory:
   - (UNIX) *ORACLE_HOME*`/oracle_common/common/bin`
   - (Windows) *ORACLE_HOME*`\oracle_common\commom\bin`

   Where, *ORACLE_HOME* is the 12c Oracle Home.
4. Start the Reconfiguration Wizard with the following logging options:
   - (UNIX) `./reconfig.sh -log=`*log_file*` -log_priority=ALL`
   - (Windows) `reconfig.cmd -log=`*log_file*` -log_priority=ALL`

   Where, *log_file* is the absolute path of the log file you'd like to create for the domain reconfiguration session. This can be helpful if you need to troubleshoot the reconfiguration process.

   The parameter `-log_priority=ALL` ensures that logs are logged in fine mode.

Reconfiguring the Oracle Identity Manager Domain

Navigate through the screens in the Reconfiguration Wizard to reconfigure your existing domain.

To reconfigure the domain with the Reconfiguration Wizard:

1. On the Select Domain screen, specify the location of the domain you want to upgrade or click **Browse** to navigate and select the domain directory. Click **Next**.
2. On the Reconfiguration Setup Progress screen, view the progress of the setup process. When complete, click **Next**.

   During this process:
   - The reconfiguration templates for your installed products, including Fusion Middleware products, are automatically applied. This updates various domain configuration files such as `config.xml`, `config-groups.xml`, and `security.xml` (among others).
   - Schemas, scripts, and other such files that support your Fusion Middleware products are updated.
   - The domain upgrade is validated.
3. On the Domain Mode and JDK screen, select the JDK to use in the domain or click **Browse** to navigate to the JDK you want to use. The supported JDK version for 14c (14.1.2.1.0) is 17.0.12 and later. Click **Next**.

> **Note:**
>
> You cannot change the **Domain Mode** at this stage.

For a list of JDKs that are supported for a specific platform, see Oracle Fusion Middleware Supported System Configurations.

4. On the Database Configuration Type screen, select **RCU Data** to connect to the Server Table (_STB) schema.

   Enter the database connection details using the RCU service table (_STB) schema credentials and click **Get RCU Configuration**.

   The Reconfiguration Wizard uses this connection to automatically configure the data sources required for components in your domain.

   > **Note:**
   >
   > By default **Oracle's Driver (Thin) for Service connections; Versions: Any** is the selected driver. If you specified an instance name in your connection details — instead of the service name — you must select **Oracle's Driver (Thin) for pooled instance connections; Versions: Any** If you do not change the driver type, then the connection will fail.

   > **Note:**
   >
   > For any existing 12c (12.2.1.4.0) datasource, the reconfiguration will preserve the existing values. For new datasources where the schema was created for 14c (14.1.2.1.0) by the RCU, the default connection data will be retrieved from the _STB schema. If no connection data for a given schema is found in the _STB schema, then the default connection data is used.

   If the check is successful, click **Next**. If the check fails, reenter the connection details correctly and try again.

   > **Note:**
   >
   > If you are upgrading from 12c (12.2.1.4.0), and your database has _OPSS or _IAU 12c (12.2.1.4.0) database schemas, you must manually enter database connection details for those schemas. These schemas were not required in 12c and had to be created manually. Users could assign any name to these schemas, therefore the Reconfiguration Wizard does not recognize them. When providing connection information for _IAU, use the IAU_APPEND user information.

5. On the JDBC Component Schema screen, verify that the DBMS/Service and the Host name is correct for each component schema and click **Next**.

> **Note:**
>
> - For all of the schemas except for OPSS, the host, port, and service details will be auto-populated. You must enter the OPSS schema credentials manually.
> - If you are using a RAC database, then on the JDBC Component Schema screen, select all the datasources and select **Convert to Grid Link**.

6. On the Grid Link screen, provide the Service Name, Schema Password, ONS Host and Port, SCAN Hostname and Port, and check the FAN and SCAN checkboxes appropriately. Also, verify that the prefix for each schema owner reflects your environment. Perform this step for each RAC Component Schema.

   When complete, click **Next**.

> **Note:**
>
> The Grid Link screen will be displayed only if you select **Convert to Grid Link** in step 6.

7. On the JDBC Component Schema Test screen, the component schema connections are tested. The result of the test is indicated in the Status column.

   When the check is complete, click **Next**.

8. On the Node Manager screen, go for the default option or select **Create New Configuration** for configuring Node Manager per your requirement. In both the cases, specify the WebLogic Administration user credentials for Node Manager details.
9. On the Credentials screen, for `weblogicAdminnKey`, populate the Weblogic admin username and password used in 11g, and then click **Next**.
10. Leave the default selection and click **Next**.
11. On the Advanced Configuration screen, you can select all categories for which you want to perform advanced configuration. For each category you select, the appropriate configuration screen is displayed to allow you to perform advanced configuration.

> **Note:**
>
> The categories that are listed on the Advanced Configuration screen depend on the resources defined in the templates you selected for the domain.

   For this upgrade, select none of the options and click **Next**.

12. On the Configuration Summary screen, review the detailed configuration settings of the domain before continuing.

    You can limit the items that are displayed in the right-most panel by selecting a filter option from the **View** drop-down list.

    To change the configuration, click **Back** to return to the appropriate screen. To reconfigure the domain, click **Reconfig**.

> **Note:**
>
> The location of the domain does not change when you reconfigure it.

13. The Reconfiguration Progress screen displays the progress of the reconfiguration process.

    During this process:

    - Domain information is extracted, saved, and updated.
    - Schemas, scripts, and other such files that support your Fusion Middleware products are updated.

    When the progress bar shows 100%, click **Next**.

14. The End of Configuration screen indicates whether the reconfiguration process completed successfully or failed. It also displays the location of the domain that was reconfigured as well as the Administration Server URL (including the listen port). If the reconfiguration is successful, it displays **Oracle WebLogic Server Reconfiguration Succeeded**.

    If the reconfiguration process did not complete successfully, an error message is displayed indicates the reason. Take appropriate action to resolve the issue. If you cannot resolve the issue, contact My Oracle Support.

    Note the Domain Location and the Admin Server URL for further operations.

Upgrading Domain Component Configurations

Use the Upgrade Assistant to upgrade the domain *component* configurations inside the domain to match the updated domain configuration.

- Starting the Upgrade Assistant
  Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 14c (14.1.2.1.0).
- Upgrading Oracle Identity Manager Domain Component Configurations
  Navigate through the screens in the Upgrade Assistant to upgrade component configurations in the WebLogic domain.

Starting the Upgrade Assistant

Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 14c (14.1.2.1.0).

To start the Upgrade Assistant:

> **Note:**
>
> Before you start the Upgrade Assistant, make sure that the JVM character encoding is set to UTF-8 for the platform on which the Upgrade Assistant is running. If the character encoding is not set to UTF-8, then you will not be able to download files containing Unicode characters in their names. This can cause the upgrade to fail. To set the character encoding, run the following:
>
> UNIX operating systems:
>
> ```
> export UA_PROPERTIES="-Dfile.encoding=UTF-8 ${UA_PROPERTIES}"
> ```
>
> Windows operating systems:
>
> ```
> set UA_PROPERTIES=-Dfile.encoding=UTF-8 %UA_PROPERTIES%
> ```

1. Go to the `oracle_common/upgrade/bin` directory:
   - (UNIX) *`ORACLE_HOME`*`/oracle_common/upgrade/bin`
   - (Windows) *`ORACLE_HOME`*`\oracle_common\upgrade\bin`
2. Start the Upgrade Assistant:
   - (UNIX) `./ua`
   - (Windows) `ua.bat`

For information about other parameters that you can specify on the command line, such as logging parameters, see:

Upgrading Oracle Identity Manager Domain Component Configurations

Navigate through the screens in the Upgrade Assistant to upgrade component configurations in the WebLogic domain.

Run the Upgrade Assistant to upgrade the domain component configurations to match the updated domain configuration.

To upgrade domain component configurations with the Upgrade Assistant:

1. On the Welcome screen, review an introduction to the Upgrade Assistant and information about important pre-upgrade tasks. Click **Next**.

   > **Note:**
   >
   > For more information about any Upgrade Assistant screen, click **Help** on the screen.

2. On the next screen:
   - Select **All Configurations Used By a Domain**. The screen name changes to WebLogic Components.
   - In the **Domain Directory** field, specify the OIM domain directory.

Where, **Domain Directory** is the Administration server domain directory.

Click **Next**.

3. On the Component List screen, verify that the list includes all the components for which you want to upgrade configurations and click **Next**.

   If you do not see the components you want to upgrade, click **Back** to go to the previous screen and specify a different domain.

4. On the Prerequisites screen, acknowledge that the prerequisites have been met by selecting all the check boxes. Click **Next**.

   > **Note:**
   >
   > The Upgrade Assistant does not verify whether the prerequisites have been met.

5. On the Examine screen, review the status of the Upgrade Assistant as it examines each component, verifying that the component configuration is ready for upgrade. If the status is **Examine finished**, click **Next**.

   If the examine phase fails, Oracle recommends that you cancel the upgrade by clicking **No** in the Examination Failure dialog. Click **View Log** to see what caused the error and refer to Troubleshooting Your Upgrade in *Upgrading with the Upgrade Assistant* for information on resolving common upgrade errors.

   > **Note:**
   >
   > - If you resolve any issues detected during the examine phase without proceeding with the upgrade, you can start the Upgrade Assistant again without restoring from backup. However, if you proceed by clicking **Yes** in the Examination Failure dialog box, you need to restore your pre-upgrade environment from backup before starting the Upgrade Assistant again.
   > - Canceling the examination process has no effect on the configuration data; the only consequence is that the information the Upgrade Assistant has collected must be collected again in a future upgrade session.

6. On the Upgrade Summary screen, review the summary of the options you have selected for component configuration upgrade.

   The response file collects and stores all the information that you have entered, and enables you to perform a silent upgrade at a later time. The silent upgrade performs exactly the same function that the Upgrade Assistant performs, but you do not have to manually enter the data again. If you want to save these options to a response file, click **Save Response File** and provide the location and name of the response file.

   Click **Upgrade** to start the upgrade process.

7. On the Upgrade Progress screen, monitor the status of the upgrade.

> **Caution:**
>
> Allow the Upgrade Assistant enough time to perform the upgrade. Do not cancel the upgrade operation unless absolutely necessary. Doing so may result in an unstable environment.

If any components are not upgraded successfully, refer to the Upgrade Assistant log files for more information.

> **Note:**
>
> The progress bar on this screen displays the progress of the current upgrade procedure. It does not indicate the time remaining for the upgrade.

Click **Next**.

8. If the upgrade is successful: On the Upgrade Success screen, click **Close** to complete the upgrade and close the wizard. The Post-Upgrade Actions window describes the manual tasks you must perform to make components functional in the new installation. This window appears only if a component has post-upgrade steps.

   If the upgrade fails: On the Upgrade Failure screen, click **View Log** to view and troubleshoot the errors. The logs are available at *ORACLE_HOME*/oracle_common/upgrade/logs.

> **Note:**
>
> If the upgrade fails you must restore your pre-upgrade environment from backup, fix the issues, then restart the Upgrade Assistant.

Tuning Application Module for User Interface

After you successfully upgrade the Oracle Identity Manager middle-tier, tune the Application Module (AM).

The parameter *jbo.ampool.maxavailablesize* is used to let OIM know the number of concurrent users expected to access OIM. To check the default value, navigate to `$DOMAIN_HOME/setDomainEnv.sh` and search for the parameter *jbo.ampool.maxavailablesize*.

If the set value does not match the number of concurrent users you expect, you need to update that value in the `setUserOverridesLate.sh` file. It is important that you do not change the `setDomainEnv.sh` file directly as changes can be lost during future updates. All user defined values should appear in `setUserOverridesLate.sh` as changes to this file are persistent across upgrades.

The recommended value for the parameter *jbo.ampool.maxavailablesize* is the number of expected concurrent Users + 20%.

To add the recommended application module settings, complete the following:

1. Open the file *$DOMAIN_HOME*/bin/setUserOverridesLate.sh in a text editor.

2. Edit the `setUserOverridesLate.sh` file to add the following line:

```
JAVA_OPTIONS="${JAVA_OPTIONS} -Djbo.ampool.maxavailablesize = <# of
concurrent users + 20%>
```

3. Save and close the `setUserOverridesLate.sh` file.

> **Note:**
>
> If the `setUserOverridesLate.sh` file does not exist, you have to create it.

Copying oracle.iam.ui.custom-dev-starter-pack.war from 12c Oracle Home

You have to manually copy the `oracle.iam.ui.custom-dev-starter-pack.war` file from the backup of 12c (12.2.1.4.0) Oracle Home to 14c (14.1.2.1.0) Oracle home: *`ORACLE_HOME`*`/idm/server/apps/`.

Copying Folders to the 14c (14.1.2.1.0) Oracle Home

When upgrading to 14*c*, you must manually copy some folders to the new Oracle Home, if those folders have file system dependent data.

For example: `plugins`, `ScheduleTask`, `XLIntegrations`, `JavaTasks`, `connectorResources`, and so on.

Run the following command:

```
cp -r 12c_MW_HOME/<product_idm>/server/plugins/* ORACLE_HOME/<product_idm>/
server/plugins/
```

Where, *`ORACLE_HOME`* is the 14*c* Oracle Home.

Starting Servers and Processes

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

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

> **Note:**
>
> The procedures in this section describe how to start servers and 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 Node Manager

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

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

Step 2: 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 3: Start Any Managed Servers

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

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

> **Note:**
>
> When using secured production mode, you must provide additional parameters to start the Managed Servers. See Starting Managed Servers using a Start Script in *Administering Security for Oracle WebLogic Server*.

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

Step 4: Start System Components

To start system components, such as Oracle HTTP Server, 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.

Verifying the Domain-Specific-Component Configurations Upgrade

To verify that the domain-specific-component configurations upgrade was successful, sign in to the Remote Console and verify that the version numbers for each upgraded component is 14.1.2.1.0.

> **Note:**
>
> Before you can access the Hosted WebLogic Remote Console, you must deploy the hosted WebLogic Remote Console. For more information, see the Remote Console Online Help.

To sign in to the Remote Console, go to: `http://hostname:port/rconsole` or for HTTPS, `https://hostname:port/rconsole`.

> **Note:**
>
> After a successful upgrade, make sure you run the administration tools from the new 14c (14.1.2.1.0) Oracle home directory and not from the previous Oracle home directory.
>
> During the upgrade process, some OWSM documents, including policy sets and predefined documents such as policies and assertion templates, may need to be upgraded. If a policy set or a predefined document is upgraded, its version number is incremented by 1.
>
> If you created the FMW user to run the Upgrade Assistant, ensure that you delete the account after verifying your upgrade was successful.

Updating the setDomainEnv.sh File

For upgrading Oracle Identity Governance (OIG) from 12c (12.2.1.4.0) to 14c (14.1.2.1.0), you need to the delete a property in the `setDomainEnv.sh` file.

Complete the following steps:

1. Open the `setDomainEnv.sh` file in the *`Oracle_Home`*`/domains/<domain name>/bin/` location.
2. Delete the following parameter from the line which starts as follows:

   ```
   EXTRA_JAVA_PROPERTIES="-Djavax.net.ssl.trustStore=${WL_HOME}/server/lib/
   DemoTrust.jks
   ```

   The parameter is:

   ```
   -Doracle.xdkjava.compatibility.version=11.1.1
   ```

3. Save and close the `setDomainEnv.sh` file.

> **Note:**
>
> - For SOA, you need to add the following entry as an argument to the `setSOADomainEnv.sh` file in the line starting with `EXTRA_JAVA_PROPERTIES="${EXTRA_JAVA_PROPERTIES}.`
>
>   ```
>   -Doracle.xdkjava.compatibility.version=11.1.1
>   ```
>
> - Repeat these steps in all the OIM host machines.

Upgrading Oracle Identity Manager Design Console

Upgrade the Oracle Identity Manager Design Console after you upgrade the Oracle Identity Manager (OIM) domain component configurations.

To upgrade the Oracle Identity Manager Design Console, complete the following steps:

1. Replace the 14c (14.1.2.1.0) `designconsole/config/xlconfig.xml` with the 12c (12.2.1.4.0) `designconsole/config/xlconfig.xml` file.
2. If the design console is not configured in the previous version, when you start the design console, the host name and port values of the OIM Managed Server are changed to default variables. In the design console's start window, update the URL to the correct values for your installation.

Post-Upgrade Tasks

After performing the upgrade of Oracle Access Manager to 14c (14.1.2.1.0), you should complete the tasks summarized in this section, if required.

This section includes the following topics:

Copying Custom Configurations

If you had set custom configuration in your 12c (12.2.1.4.0) Oracle home, you need to copy the custom configuration present in your backup of 12c (12.2.1.4.0) Oracle home to the 14c (14.1.2.1.0) Oracle home.

For example: Copy any contents from standard directories such as `XLIntegrations`, `connectorResources`, and so on, under the backup of 12c (12.2.1.4.0) Oracle home to the corresponding directories under the 14c (14.1.2.1.0) Oracle home.

Similarly, if your schedule job parameters are referring anything from the 12c (12.2.1.4.0) Oracle home, then copy them from the backup of 12c (12.2.1.4.0) Oracle home to the corresponding directories under the 14c (14.1.2.1.0) Oracle home.

> **Note:**
>
> The back up of custom configurations that you created in Backing up the 12c (12.2.1.4.0) Oracle Home Folder on OIMHOST are restored in this step.

Handling Custom Applications

If custom applications and libraries are present in your deployment of Oracle Identity Governance (OIG) 12c (12.2.1.4.0), Oracle recommends you to update them manually after the upgrade to Oracle Identity Governance (OIG) 14c (14.1.2.1.0).

Reinstalling the ADF DI Excel Plug-in

After you upgrade Oracle Identity Manager to 14c (14.1.2.1.0), uninstall and reinstall the ADF DI Excel plug-in, and then re-download the Excel.

Defining System Properties for Legacy Connectors

As part of post-upgrade tasks, for legacy connectors such as Resource Access Control Facility (RACF) that use the `tcITResourceInstanceOperationsBean.getITResourceInstanceParameters` method, you should create the following two system properties and update their values to `True`:

- `Service Account Encrypted Parameter Value`
- `Service Account Parameters Value Store`

For more information about these system properties, see Table 18-2 of section Non-Default System Properties in Oracle Identity Governance in *Administering Oracle Identity Governance*.

Oracle recommends creating these system properties only if a legacy connector or an old custom code requires the legacy behavior.

Increasing the Maximum Message Size for WebLogic Server Session Replication

Oracle recommends you to modify the Maximum Message Size from the default value of 10 MB to 100 MB. This value is used to replicate the session data across the nodes. You should perform this step for all the Managed servers and the Administration server.

1. Log in to the WebLogic Server Administration Console.
2. Navigate to **Servers**, select **Protocols**, and then click **General**.
3. Set the value of **Maximum Message Size** to `100 MB`.

Increasing the `maxdepth` Value in setDomainEnv.sh

The recommended value for the `maxdepth` parameter is `250`. To update this value:

1. Open the `$DOMAIN_HOME/bin/setDomainEnv.sh` file in a text editor.
2. Locate the following code block:

```
ALT_TYPES_DIR="${OIM_ORACLE_HOME}/server/loginmodule/wls,$
{OAM_ORACLE_HOME}/a
gent/modules/oracle.oam.wlsagent_11.1.1,${ALT_TYPES_DIR}"
export ALT_TYPES_DIR
CLASS_CACHE="true"
export CLASS_CACHE
```

3. Add the following lines at the end of the above code block:

```
JAVA_OPTIONS="${JAVA_OPTIONS} -Dweblogic.oif.serialFilter=maxdepth=250"
export JAVA_OPTIONS
```

4. Save and close the `setDomainEnv.sh` file.

Changing the JMS and TLOG Persistence Store After the Upgrade

The JMS and TLOG persistent store remain the same after the upgrade to Oracle Identity Manager 14c (14.1.2.1.0). That is, if the persistence store is file-based prior to the upgrade, it will be file-based after the upgrade as well.

If you want to change the persistence stores from a file-based system to a database-based system, you have to perform the steps manually. See Using Persistent Stores for TLOGs and JMS in an Enterprise Deployment.

4

Upgrading Oracle Identity Manager Highly Available Environments

This chapter describes the process of upgrading an Oracle Identity Manager highly available environment from 12c (12.2.1.4.0) to Oracle Identity Governance 14c (14.1.2.1.0).

> **Note:**
>
> - You can use the rolling upgrade process to upgrade a highly available environment from 12c (12.2.1.4.0) to 14c (14.1.2.1.0) with zero downtime.
> - In this chapter, Oracle Identity Manager (OIM) and Oracle Identity Governance (OIG) are used interchangeably.

Topics

- Upgrading Product Schemas From OIMHOST1
  Upgrade all of the necessary schemas for Oracle Identity Manager, from OIMHOST1 by using the Upgrade Assistant.
- Upgrading Domain Component Configurations on OIMHOST1
  Use the Upgrade Assistant to upgrade the domain component's configurations inside the domain to match the updated domain configuration.
- Verifying the Domain-Specific-Component Configurations Upgrade
  To verify that the domain-specific-component configurations upgrade was successful, sign in to the Administration console and the Oracle Enterprise Manager Fusion Middleware Control and verify that the version numbers for each component is 14.1.2.1.0.
- Updating the setDomainEnv.sh File
  For upgrading Oracle Identity Governance (OIG) from 12c (12.2.1.4.0) to 14c (14.1.2.1.0), you need to the delete a property in the `setDomainEnv.sh` file.
- Performing OIM Bootstrap on OIMHOST1
  After you upgrade Oracle Identity Manager on OIMHOST1, restart the servers.
- Handling Custom Applications
- Packing Domain Configurations on OIMHOST1
  After upgrading domain component configurations on OIMHOST1, pack the upgraded domain on OIMHOST1. You must unpack it later on OIMHOST2.
- Starting Servers and Processes
  After a successful upgrade, shut down any servers you may have started manually, and then restart all processes and servers, including the Administration Server and any Managed Servers.
- Stopping Servers and Processes on OIMHOST2
  Before you upgrade the schemas and configurations, you must shut down all of the pre-upgrade processes and servers, including the Administration Server, Node Manager, and any managed servers on OIMHOST2.
- Upgrading the Binaries on OIMHOST2
  You have to perform these steps only if OIMHOST2 is using a different binary location as compared to that of OIMHOST1.
- Replicating the Domain Configurations on Each OIMHOST
  Replicate the domain configurations on OIMHOST2. This involves unpacking the upgraded domain on OIMHOST2, which was packed on OIMHOST1.
- Deploy the oracle.iam.ui.custom-dev-starter-pack.war
  Validate that the Upgrade Assistant has automatically copied the `oracle.iam.ui.custom-dev-starter-pack.war` file from the 12c (12.2.1.4.0) *MW_HOME* to the 14c (14.1.2.1.0) *ORACLE_HOME* on the AdminServer host.
- Starting the Servers on OIMHOST2
  After you upgrade Oracle Identity Manager on OIMHOST2, restart the servers.
- Post-Upgrade Task
  After performing the upgrade of Oracle Access Manager to 14c (14.1.2.1.0), you should complete the tasks summarized in this section, if required.

About the Oracle Identity Manager Multinode Upgrade Process

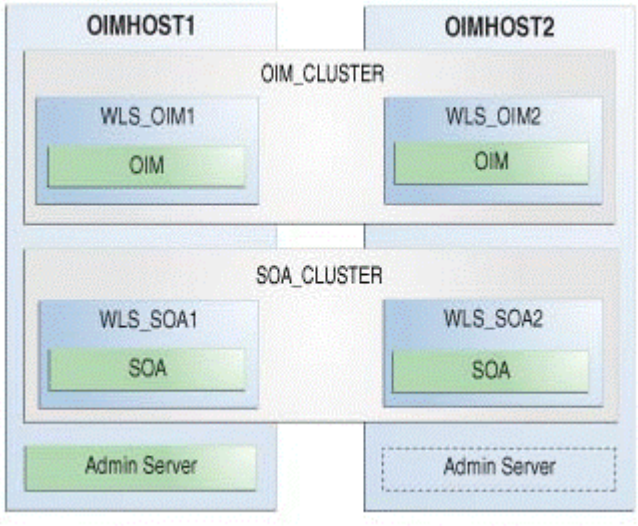
Review the topology and the roadmap for an overview of the upgrade process for Oracle Identity Manager highly available environments.

The steps you take to upgrade your existing domain will vary depending on how your domain is configured and which components are being upgraded. Follow only those steps that are applicable to your deployment.

Upgrade Topology

The following topology shows the Oracle Identity Manager cluster setup that can be upgraded to 14c (14.1.2.1.0) by following the procedure described in this chapter.

Figure 4-1 Oracle Identity Manager High Availability Upgrade Topology



On OIMHOST1, the following installations have been performed:

- An Oracle Identity Manager instance has been installed in the WLS_OIM1 Managed Server and a SOA instance has been installed in the WLS_SOA1 Managed Server.
- A WebLogic Server Administration Server has been installed. Under normal operations, this is the active Administration Server.

On OIMHOST2, the following installations have been performed:

- An Oracle Identity Manager instance has been installed in the WLS_OIM2 Managed Server and a SOA instance has been installed in the WLS_SOA2 Managed Server.
- A WebLogic Server Administration Server has been installed. Under normal operations, this is the passive Administration Server. You make this Administration Server active if the Administration Server on OIMHOST1 becomes unavailable.

The instances in the WLS_OIM1 and WLS_OIM2 Managed Servers on OIMHOST1 and OIMHOST2 are configured as the OIM_CLUSTER cluster.

The instances in the WLS_SOA1 and WLS_SOA2 Managed Servers on OIMHOST1 and OIMHOST2 are configured as the SOA_CLUSTER cluster.

Performing a Rolling Upgrade

In Oracle 14c (14.1.2.1.0), it is possible to perform a rolling upgrade to minimise your downtime. This is possible only if:

- Each host in your topology uses a local binary installation.
- You use multiple redundant binary installations on a shared storage.

If either of the above conditions is true, you can upgrade the hosts associated with each binary installation independently, that is, have a few Managed servers running Oracle Identity Manager 12c (12.2.1.4.0) while others use Oracle Identity Manager14c (14.1.2.1.0).

> **Note:**
>
> If you are following this methodology, you must not use the OIM system Administration Console until all members of the cluster are running on the same version.

Considerations for a Rolling Upgrade:

Prior to upgrade, move OIM applications session from the *replicated_if_clustered* mode to the *memory* mode. In this setting, failover of one node will not be handled by other node. If a node crashes, all users session on the node would be lost. You need to log in and perform the operations again, which were in progress when the node crashed.

Complete the following steps to move the OIM applications session from the *replicated_if_clustered* mode to the *memory* mode, on all the nodes:

1. In the binary installation Oracle Home 12c (12.2.1.4.0), change the session descriptor, from *replicated_if_clustered* to *memory*, for the following files:
   - `<12c_oracle_home>/idm/server/apps/oim.ear/xlWebApp.war/WEB-INF/weblogic.xml`
   - `<12c_oracle_home>/idm/server/apps/oim.ear/iam-consoles-faces.war/WEB-INF/weblogic.xml`

   For example: change from

   ```
   <session-descriptor>
   <persistent-store-type>replicated_if_clustered</persistent-store-type>
             <cookie-name>oimjsessionid</cookie-name>
       <url-rewriting-enabled>false</url-rewriting-enabled>
     </session-descriptor>
   ```

   to

   ```
   <session-descriptor>
           <persistent-store-type>memory</persistent-store-type>
             <cookie-name>oimjsessionid</cookie-name>
       <url-rewriting-enabled>false</url-rewriting-enabled>
     </session-descriptor>
   ```

2. Restart all Managed servers that you have changed in step 1.
3. On Node 1, after installing the 14c (14.1.2.1.0) binaries, change the session descriptor, from *replicated_if_clustered* to *memory*, for file: `<12c_oracle_home>/idm/server/apps/oim.ear/iam-consoles-faces.war/WEB-INF/weblogic.xml`

> **Note:**
>
> xlWebApp is not present in 14c (14.1.2.1.0) binaries.

4. Proceed with the upgrade of the WebLogic Administration Server followed by the upgrade of each Managed Server that is running from the *Oracle_Home* you are upgrading. After completing, continue upgrading the Managed Servers associated with other *Oracle_Home* installations.

> **Note:**
>
> In this case, *Oracle_Home* refers to the installation of the Oracle binaries you are using to upgrade. Upgrade a node at a time if you are using the local binary installations, Or upgrade all the nodes associated with a shared storage binary installation if you are using redundant shared storage installations.

5. After upgrading all the nodes to 14c (14.1.2.1.0), you can switch again to the `replicated_if_clustered` mode.

Table 4-1 Roadmap for Upgrading Oracle Identity Manager Highly Available Environments

Task	Description
Required If you have not done so already, review the introductory topics in this guide and complete the required pre-upgrade tasks.	See: • Introduction to Upgrading Oracle Identity and Access Management to 14c (14.1.2.1.0) • Pre-Upgrade Requirements
Required Complete the necessary pre-upgrade tasks specific to Oracle Identity Manager.	See Completing the Pre-Upgrade Tasks for Oracle Identity Manager.
Required on OIMHOST1 Shutdown the 12c servers running from the Oracle Home you are upgrading. This includes the Administration Server, Managed Servers, Node Manager, and system components such as Oracle HTTP Server. Ensure that the Database is up during the upgrade.	**WARNING**: Failure to shut down your servers during an upgrade may lead to data corruption. See Stopping Servers and Processes.
Required Create backup of the existing 12c (12.2.1.4.0) Oracle home folders on OIMHOSTs	See Backing up the 12c (12.2.1.4.0) Oracle Home Folder on OIMHOSTs. **Note:** Backup any UI customizations made in 12c (12.2.1.4.0), which is the `oracle.iam.ui.custom-dev-starter-pack.war` file.
Required on OIMHOST1 On OIMHOST1, uninstall Oracle Fusion Middleware Infrastructure and Oracle Identity Manager 12c (12.2.1.4.0) in the existing Oracle home.	See Uninstalling the Software on OIMHOST1.

Table 4-1 (Cont.) Roadmap for Upgrading Oracle Identity Manager Highly Available Environments

Task	Description
Required on OIMHOST1 On OIMHOST1, install Infrastructure (JRF) 14*c* (14.1.2.0.0), Oracle SOA Suite, and Oracle Identity Manager 14c (14.1.2.1.0) in the Oracle home.	See Installing Product Distributions on an OIMHOST.
Required on OIMHOST1 Update the JDK location	See Updating the JDK Location On OIMHOST1.
Optional Run a pre-upgrade readiness check.	See Running a Pre-Upgrade Readiness Check.
Required on OIMHOST1 Upgrade the necessary schemas on OIMHOST1.	See Upgrading Schemas on OIMHOST1.
Required on OIMHOST1 Upgrade the Oracle Identity Manager configurations on OIMHOST1, using the Upgrade Assistant.	The Upgrade Assistant is used to update the domain's component configurations. See Upgrading Domain Component Configurations. **Note:** The `jce` should use unlimited strength crypto policy.
Required Verify that the domain-specific-component configurations is successful.	See Verifying the Domain-Specific-Component Configurations Upgrade.
Required on OIMHOST1 Update the `setDomainEnv.sh` file.	See Updating the setDomainEnv.sh File.
Required on OIMHOST1 Perform the bootstrap after the upgrade.	See Performing OIM Bootstrap on OIMHOST1
Required on OIMHOST1 Handle custom applications.	See Handling Custom Applications.
Required on OIMHOST1 Pack the domain on OIMHOST1.	See Packing Domain Configurations on OIMHOST1.
Required on OIMHOST1 After a successful upgrade, restart all processes and servers.	See Starting Servers and Processes.
Required on OIMHOST2 Shutdown the servers on other cluster nodes, if present. This includes the SOA server, OIM server, and Node Manager. Ensure that the Database is up during the upgrade.	**WARNING**: Failure to shut down your servers during an upgrade may lead to data corruption. See Stopping Servers and Processes on OIMHOST2.
Optional Upgrade the binaries on OIMHOST2.	See Upgrading the Binaries on OIMHOST2.
Required on OIMHOST2 Replicate the domain configurations on OIMHOST2, and to each host being serviced by the Oracle Home you are upgrading.	This includes unpacking the domain on OIMHOST2. See Replicating the Domain Configurations on Each OIMHOST.

Table 4-1 (Cont.) Roadmap for Upgrading Oracle Identity Manager Highly Available Environments

Task	Description
Required on all hosts Copy the `oracle.iam.ui.custom-dev-starter-pack.war` file to 14c (14.1.2.1.0) Oracle Home on all hosts.	See Deploy the `oracle.iam.ui.custom-dev-starter-pack.war`
Required on OIMHOST2 Start the servers in the recommended order. Also, ensure that each server is started and running before starting the next server.	See Starting the Servers on OIMHOST2.
Optional Perform the post-upgrade tasks.	See Post-Upgrade Task.

> **Note:**
>
> Repeat all the steps performed on OIMHOST2, on the other nodes in your HA environment.

Completing the Pre-Upgrade Tasks for Oracle Identity Manager

Complete the pre-upgrade tasks described in this section before you upgrade Oracle Identity Manager.

- Verifying the Memory Settings
  To avoid the memory issues for Oracle Identity Manager, ensure that the memory settings are updated as per the requirements.
- Opening the Non-SSL Ports for SSL Enabled Setup
  If you have an SSL enabled and non-SSL disabled setup, you must open the non-SSL ports for the database before you proceed with the Oracle Identity Manager upgrade.
- Clean Temporary Folder
  Clean the `/tmp` folder on all the Oracle Identity Governance host machines.
- Backing Up the metadata.mar File Manually

Verifying the Memory Settings

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

On Linux, as a `root` user, do the following:

1. Ensure that you set the following parameters in the `/etc/security/limits.conf` or `/etc/security/limits.d` file, to the specified values:

   ```
   FUSION_USER_ACCOUNT soft nofile 32767
   FUSION_USER_ACCOUNT hard nofile 327679
   ```

2. Ensure that you set `UsePAM` to `Yes` in the `/etc/ssh/sshd_config` file.
3. Restart `sshd`.

4. Check the `maxproc` limit and increase it to a minimum of 16384, if needed. Increasing the limit will ensure you do not run into memory issues.

   Use the following command to check the limit:

   ```
   ulimit -u
   ```

   If less than 16384, use following command to increase the limit of open files:

   ```
   ulimit -u 16384
   ```

   > **Note:**
   >
   > You can verify that the limit has been set correctly by reissuing the command `ulimit -u`.

   To ensure that the settings persist at reboot, add the following line to the `/etc/security/limits.conf` file or `/etc/security/limits.d` file:

   ```
   oracle hard nproc 16384
   ```

   Where, `oracle` is the install user.

5. Log out (or reboot) and log in to the system again.

Opening the Non-SSL Ports for SSL Enabled Setup

If you have an SSL enabled and non-SSL disabled setup, you must open the non-SSL ports for the database before you proceed with the Oracle Identity Manager upgrade.

Ensure that the database listener is listening on the same TCP port for the database servers that you provided to Upgrade Assistant as parameters. For more information, see Enabling SSL for Oracle Identity Governance DB.

Clean Temporary Folder

Clean the `/tmp` folder on all the Oracle Identity Governance host machines.

As the `/tmp` directory is set against the JVM `java.io.tmpdir` property, any unwanted files in the `/tmp` folder can interfere with OIG upgrade process and might result is MDS corruption.

Backing Up the `metadata.mar` File Manually

After you install the 14c (14.1.2.1.0) binaries in the existing Oracle Home, take a backup of the 14c (14.1.2.1.0)_*ORACLE_HOME*>`/idm/server/apps/oim.ear/metadata.mar` file before the upgrade.

Stopping Servers and Processes on OIMHOST1

Before you upgrade the schemas and configurations, you must shut down all of the pre-upgrade processes and servers, including the Administration Server, Node Manager, and any Managed servers on OIMHOST1, running out of the Oracle Home you are upgrading.

An Oracle Fusion Middleware environment can consist of an Oracle WebLogic Server domain, an Administration Server, multiple managed servers, Java components, system components such as Identity Management components, and a database used as a repository for metadata. The components may be dependent on each other, so they must be stopped in the correct order.

> **Note:**
>
> - The procedures in this section describe how to stop the existing, pre-upgrade 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 Administration Console. See Starting and Stopping Administration and Managed Servers and Node Manager.
> - Stop all of the servers in your deployment, except for the Database. The Database must be up during the upgrade process.

To stop your pre-upgrade Fusion Middleware environment, navigate to the pre-upgrade domain and follow the steps below.

Step 1: Stop the Managed Servers

Depending on the method you followed to start the managed servers, follow one of the following methods to stop the WebLogic Managed Server:

Method 1: To stop a WebLogic Server Managed Server not managed by Node Manager:

- (UNIX) `DOMAIN_HOME/bin/stopManagedWebLogic.sh managed_server_name admin_url`
- (Windows) `DOMAIN_HOME\bin\stopManagedWebLogic.cmd managed_server_name admin_url`

When prompted, enter your user name and password.

Method 2: To stop a WebLogic Server Managed Server by using the Weblogic Console:

- Log into Weblogic console as a `weblogic` Admin.
- Go to **Servers** > **Control** tab.
- Select the required managed server.
- Click **Shutdown**.

Method 3: To stop a WebLogic Server Managed Server using node manager, run the following commands:

```
wls:/offline>nmConnect('nodemanager_username','nodemanager_password',
            'AdminServerHostName','5556','domain_name',
            'DOMAIN_HOME','nodemanager_type')
```

```
wls:/offline>nmKill('ManagedServerName')
```

Step 2: Stop the Administration Server

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

Follow one of the these methods to stop the Administration Server:

Method 1: To stop the Administration Server not managed by Node Manager:

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

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

Method 2: To stop the Administration Server by using the Weblogic Console:

- Log into Weblogic console as a `weblogic` Admin.
- Go to **Servers** > **Control** tab.
- Select the required admin server.
- Click **Shutdown**.

Method 3: To stop a WebLogic Server Managed Server using Node Manager, run the following commands:

```
wls:/offline>nmConnect('nodemanager_username','nodemanager_password',
            'AdminServerHostName','5556','domain_name',
            'DOMAIN_HOME','nodemanager_type')

wls:/offline>nmKill('AdminServer')
```

Step 3: Stop Node Manager

To stop Node Manager, run the following command:

```
<DOMAIN_HOME>/bin/stopNodeManager.sh
```

Backing up the 12c (12.2.1.4.0) Oracle Home Folder on OIMHOSTs

Backup the 12c (12.2.1.4.0) Oracle Home on both OIMHOST1 and OIMHOST2.

As a backup, copy and rename the 12.2.1.4.0 Oracle Home folder on OIMHOST1 and OIMHOST2.
For example:

From `/u01/app/fmw/ORACLE_HOME` to `/u01/app/fmw/ORACLE_HOME_old`

> **Note:**
>
> Ensure that you back up any custom configuration. Post upgrade, you will restore these configurations.

Uninstalling the Software on OIMHOST1

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*.

Follow these steps to uninstall the software:

- Starting the Uninstall Wizard
- Selecting the Product to Uninstall
- Navigating the Uninstall Wizard Screens

Starting the Uninstall Wizard

Start the Uninstall Wizard:

1. Change to the following directory:

   (UNIX) *ORACLE_HOME*`/oui/bin`

   (Windows) *ORACLE_HOME*`\oui\bin`
2. Enter the following command:

   (UNIX) `./deinstall.sh`

   (Windows) `deinstall.cmd`

Selecting the Product to Uninstall

Because multiple products exist in Oracle Home, ensure that you uninstall each product. You will be installing the latest product distribution in this location. The installer requires the directory to be empty.

When you launch the Uninstall Wizard, the Distribution to Uninstall screen opens.

From the drop-down menu, select the **Oracle Fusion Middleware 14c (14.1.2.1.0) Identity and Access Management 14c (14.1.2.1.0)** product and click **Uninstall**.

After the uninstallaion is complete, redo the uninstall process for each additional product in Oracle Home, until all products are removed.

The uninstallation program shows the screens listed in Navigating the Uninstall Wizard Screens.

Navigating the Uninstall Wizard Screens

The Uninstall Wizard shows a series of screens to confirm the removal of the software.

If you need help on screen listed in the following table, click **Help** on the screen.

Table 4-2 Uninstall Wizard Screens and Descriptions

Screen	Description
Welcome	Introduces you to the product Uninstall Wizard.
Uninstall Summary	Shows the Oracle home directory and its contents that are uninstalled. Verify that this is the correct directory. If you want to save these options to a response file, click **Save Response File** and enter the response file location and name. You can use the response file later to uninstall the product in silent (command-line) mode. See Running the Oracle Universal Installer for Silent Uninstall in *Installing Software with the Oracle Universal Installer*. Click **Deinstall**, to begin removing the software.
Uninstall Progress	Shows the uninstallation progress.
Uninstall Complete	Appears when the uninstallation is complete. Review the information on this screen, then click **Finish** to close the Uninstall Wizard.

> **Note:**
>
> For installations that have `user_projects` Domain Home information in the *ORACLE_HOME* directory: Delete all files and directories under the <*OIM_HOME*> except for the `user_projects` directory and `domain-registry.xml` file.
> For installations that have `user_projects` Domain Home information a different directory than the *ORACLE_HOME*: Delete all files and directories under the <*OIM_HOME*> except the `domain-registry.xml` file.

After uninstalling the product, manually remove the *ORACLE_HOME* and any remaining files. If you do not empty the directory, you cannot proceed with installation.

Installing Product Distributions on OIMHOST1

After you have uninstalled the software from the 12c (12.2.1.4.0) Oracle home, install the 14c (14.1.2.1.0) binaries into the same Oracle home.

Install the following products on OIMHOST1:

- Oracle Fusion Middleware Infrastructure 14*c* (14.1.2.0.0)
- Oracle SOA Suite
- Oracle Identity Manager 14c (14.1.2.1.0)

> **Note:**
>
> If you have unistalled the product from a shared storage, you need to reinstall it into a shared storage and any redundant locations. If you have uninstalled the product from each OIM host, you need to reinstall it on each OIM host.

- Installing Product Distributions
  Before beginning your upgrade, download Oracle Fusion Middleware Infrastructure 14*c* (14.1.2.0.0), Oracle SOA Suite 14*c* (14.1.2.0.0), and Oracle Identity Manager 14c (14.1.2.1.0) distributions on the target system and install them by using the following commands, in the existing 12c (12.2.1.4.0) Oracle Home.

Installing Product Distributions

Before beginning your upgrade, download Oracle Fusion Middleware Infrastructure 14*c* (14.1.2.0.0), Oracle SOA Suite 14*c* (14.1.2.0.0), and Oracle Identity Manager 14c (14.1.2.1.0) distributions on the target system and install them by using the following commands, in the existing 12c (12.2.1.4.0) Oracle Home.

Note:

- Ensure that you have installed Java Development Kit (JDK) jdk17.0.12 or later on all the nodes hosting Oracle Identity Manager.
- If the `user_projects` directory and the `domain-registry.xml` file are left in place in *`ORACLE_HOME`*, you should use the `-novalidation` option to prevent the installation from failing.
  Following is an example of the failure message:

```
Verifying data......
[VALIDATION] [ERROR]:INST-07319: Validation of Oracle Home
location failed. The location specified already exists and is a
nonempty directory and not a valid Oracle Home
[VALIDATION] [SUGGESTION]:Provide an empty or nonexistent
directory location, or a valid existing Oracle Home
installation Failed. Exiting installation due to data validation
failure.
he Oracle Universal Installer failed. Exiting.
```

Note:

When Infrastructure is required for the upgrade, you must install the Oracle Fusion Middleware distribution first before you install other Fusion Middleware products.

It is recommended that you use the simplified installation process to install the products mentioned above, using the quickstart installer (`fmw_14.1.2.1.0_idmquickstart.jar`). The quickstart installer installs the Infrastructure, Oracle SOA Suite, and Oracle Identity Manager 14c (14.1.2.1.0) in one go.

Note:

If you are using Redundant binary locations, ensure that you install the software into each of those redundant locations.

See Installing Oracle Identity Governance Using Quickstart Installer in the *Installing and Configuring Oracle Identity and Access Management*.

The other option is to install the required product distributions - Infrastructure, Oracle SOA Suite, and Oracle Identity Manager 14c (14.1.2.1.0) separately. To do this, complete the following steps:

1. Sign in to the target system (OIMHOST1).
2. Download the following from Oracle Technology Network or Oracle Software Delivery Cloud to your target system:
   - If you not yet installed Oracle Fusion Middleware Infrastructure, then download Oracle Fusion Middleware Infrastructure (`fmw_14.1.2.0.0_infrastructure.jar`)
   - Oracle SOA Suite (`fmw_14.1.2.0.0_soa.jar`)
   - Oracle Identity and Access Management 14.1.2.1.0 (`fmw_14.1.2.1.0_idm_Disk1_1of1.zip`, which contains `fmw_14.1.2.1.0_idm.jar`) from OTN or Oracle Fusion Middleware 14c (14.1.2.1.0) Identity and Access Management from Oracle Software Delivery Cloud.

   > **Note:**
   >
   > Ensure that the *ORACLE_HOME* folder exists and it does not contain any files or folders. If there are any remaining files or folders in the *ORACLE_HOME* folder, delete them.

3. Change to the directory where you downloaded the 14c (14.1.2.1.0) product distribution.
4. If you have already installed Oracle Fusion Middleware Infrastructure (`fmw_14.1.2.0.0_infrastructure.jar`), go to step 15.
5. Start the installation program for Oracle Fusion Middleware Infrastructure pointing to the new JDK. Pointing to the new JDK location helps to skip a step later in the upgrade process.

   Run the following commands:
   - (UNIX) *NEW_JDK_HOME*`/bin/java -jar fmw_14.1.2.0.0_infrastructure.jar`
   - (Windows) *NEW_JDK_HOME*`\bin\java -jar fmw_14.1.2.0.0_infrastructure.jar`
6. On UNIX operating systems, the Installation Inventory Setup screen appears if this is the first time you are installing an 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, and click **Next**.

   > **Note:**
   >
   > The Installation Inventory Setup screen does not appear on Windows operating systems.

7. On the Welcome screen, review the information to make sure that you have met all the prerequisites. Click **Next**.
8. On the Auto Updates screen, select an option:

- **Skip Auto Updates**: If you do not want your system to check for software updates at this time.
- **Select patches from directory**: To navigate to a local directory if you downloaded patch files.
- **Search My Oracle Support for Updates**: To automatically download software updates if you have a My Oracle Support account. You must enter Oracle Support credentials then click **Search**. To configure a proxy server for the installer to access My Oracle Support, click **Proxy Settings**. Click **Test Connection** to test the connection.

Click **Next**.

9. On the Installation Location screen, specify the location for the existing 12c (12.2.1.4.0) Oracle home directory and click **Next**.

   For example: If 12c (12.2.1.4.0) *Oracle_home* is located under `/u01/app/fmw/`, first uninstall 12c (12.2.1.4.0) and clean up the directory to install 14c (14.1.2.1.0) into `/u01/app/fmw/`.

   For more information about Oracle Fusion Middleware directory structure, see Understanding Directories for Installation and Configuration in *Oracle Fusion Middleware Planning an Installation of Oracle Fusion Middleware*.

10. On the Installation Type screen, select **Fusion Middleware Infrastructure**.

    Click **Next**.

11. The Prerequisite Checks screen analyzes the host computer to ensure that the specific operating system prerequisites have been met.

    To view the list of tasks that are 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).

12. On the Installation Summary screen, verify the installation options that 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.

13. On the Installation Progress screen, when the progress bar displays 100%, click **Finish** to dismiss the installer, or click **Next** to see a summary.

14. The Installation Complete screen displays the Installation Location and the Feature Sets that are installed. Review this information and click **Finish** to close the installer.

15. After you have installed Oracle Fusion Middleware Infrastructure, enter the following command to start the installer for your product distribution and repeat the steps above to navigate through the installer screens:

    For installing Oracle SOA Suite 14*c* (14.1.2.0.0), run the following installer:

    > **Note:**
    >
    > On the Installation Type screen, for Oracle SOA Suite, select **Oracle SOA Suite**.

    - (UNIX) *NEW_JDK_HOME*`/bin/java -jar fmw_14.1.2.0.0_soa.jar`

- (Windows) *NEW_JDK_HOME*\bin\java -jar fmw_14.1.2.0.0_soa.jar

For installing Oracle Identity Manager 14c (14.1.2.1.0), run the following installer:

> **Note:**
>
> On the Installation Type screen, for Oracle Identity Manager, select **Collocated Oracle Identity and Access Manager**.

- (UNIX) *NEW_JDK_HOME*/bin/java -jar fmw_12.2.1.3.0_idm.jar
- (Windows) *NEW_JDK_HOME*\bin\java -jar fmw_12.2.1.3.0_idm.jar

16. If your existing 12c (12.2.1.4.0) DOMAIN_HOME resides within the 12c (12.2.1.4.0) Oracle home directory, do the following:

> **Note:**
>
> You need to perform this step only on OIMHOST1.

a. Go to the 12c (12.2.1.4.0) Oracle home backup location.
For example: /u01/app/fmw/*ORACLE_HOME_old*/

b. Copy the user_projects folder.

c. Go to the new installed 14c (14.1.2.1.0) Oracle home location.
For example: /u01/app/fmw/*ORACLE_HOME*/

d. Paste the copied user_projects folder.

Updating the JDK Location On OIMHOST1

When upgrading from 12c (12.2.1.4.0) to 14c (14.1.2.1.0), the reconfiguration wizard is not used. So, the latest JDK version is not automatically updated in the domain home.

After upgrading to 14c (14.1.2.1.0), you must search the references to the current JDK in domain home and replace those instances with the location of the new JDK.

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

Complete the following steps to manually search and replace the JDK instances:

1. Change directory to the *DOMAIN_HOME* location.
2. By using grep commands, search the *DOMAIN_HOME* for files containing the old JDK version. The following example excludes logs ending in .log and .out, .txt, and .csv files.

```
$ grep -rl <OLD_JDK_VERSION> * | grep -v "\.log" | grep -v "\.txt" | grep -v
"\.csv" | grep -v "\.out"
```

For more information about updating the JDK location, see Updating the JDK Location in an Existing Domain Home.

Running a Pre-Upgrade Readiness Check

To identify potential issues with the upgrade, Oracle recommends that you run a readiness check before you start the upgrade process. Be aware that the readiness check may not be able to discover all potential issues with your upgrade. An upgrade may still fail, even if the readiness check reports success.

- About Running a Pre-Upgrade Readiness Check
  You can run the Upgrade Assistant in `-readiness` mode to detect issues before you perform the actual upgrade. You can run the readiness check in GUI mode using the Upgrade Assistant or in silent mode using a response file.
- Starting the Upgrade Assistant in Readiness Mode
  Use the `-readiness` parameter to start the Upgrade Assistant in readiness mode.
- Performing a Readiness Check with the Upgrade Assistant
  Navigate through the screens in the Upgrade Assistant to complete the pre-upgrade readiness check.
- Understanding the Readiness Report
  After performing a readiness check for your domain, review the report to determine whether you need to take any action for a successful upgrade.

About Running a Pre-Upgrade Readiness Check

You can run the Upgrade Assistant in `-readiness` mode to detect issues before you perform the actual upgrade. You can run the readiness check in GUI mode using the Upgrade Assistant or in silent mode using a response file.

The Upgrade Assistant readiness check performs a read-only, pre-upgrade review of your Fusion Middleware schemas and WebLogic domain configurations that are at a supported starting point. The review is a read-only operation.

The readiness check generates a formatted, time-stamped readiness report so you can address potential issues before you attempt the actual upgrade. If no issues are detected, you can begin the upgrade process. Oracle recommends that you read this report thoroughly before performing an upgrade.

You can run the readiness check while your existing Oracle Fusion Middleware domain is online (while other users are actively using it) or offline.

You can run the readiness check any number of times before performing any actual upgrade. However, do not run the readiness check after an upgrade has been performed, as the report results may differ from the result of pre-upgrade readiness checks.

> **Note:**
>
> To prevent performance from being affected, Oracle recommends that you run the readiness check during off-peak hours.

Starting the Upgrade Assistant in Readiness Mode

Use the `-readiness` parameter to start the Upgrade Assistant in readiness mode.

To perform a readiness check on your pre-upgrade environment with the Upgrade Assistant:

1. Go to the `oracle_common/upgrade/bin` directory:
   - (UNIX) *ORACLE_HOME*`/oracle_common/upgrade/bin`
   - (Windows) *ORACLE_HOME*`\oracle_common\upgrade\bin`

   Where, *ORACLE_HOME* is the 14c (14.1.2.1.0) Oracle Home.
2. Start the Upgrade Assistant.
   - (UNIX) `./ua -readiness`
   - (Windows) `ua.bat -readiness`

> **Note:**
>
> If the `DISPLAY` environment variable is not set up properly to allow for GUI mode, you may encounter the following error:
>
> ```
> Xlib: connection to ":1.0" refused by server
> Xlib: No protocol specified
> ```
>
> To resolve this issue you need to set the `DISPLAY` variable to the host and desktop where a valid `X` environment is working.
>
> For example, if you are running an `X` environment inside a VNC on the local host in desktop 6, then you would set `DISPLAY=:6`. If you are running `X` on a remote host on desktop 1 then you would set this to `DISPLAY=remoteHost:1`.

For information about other parameters that you can specify on the command line, see:

- Upgrade Assistant Parameters

Upgrade Assistant Parameters

When you start the Upgrade Assistant from the command line, you can specify additional parameters.

Table 4-3 Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-readiness`	Required for readiness checks **Note**: Readiness checks cannot be performed on standalone installations (those not managed by the WebLogic Server).	Performs the upgrade readiness check without performing an actual upgrade. Schemas and configurations are checked. Do not use this parameter if you have specified the `-examine` parameter.

Table 4-3 (Cont.) Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-threads`	Optional	Identifies the number of threads available for concurrent schema upgrades or readiness checks of the schemas. The value must be a positive integer in the range 1 to 8. The default is 4.
`-response`	Required for silent upgrades or silent readiness checks	Runs the Upgrade Assistant using inputs saved to a response file generated from the data that is entered when the Upgrade Assistant is run in GUI mode. Using this parameter runs the Upgrade Assistant in *silent mode* (without displaying Upgrade Assistant screens).
`-examine`	Optional	Performs the examine phase but does not perform an actual upgrade. Do not specify this parameter if you have specified the `-readiness` parameter.
`-logLevel` *attribute*	Optional	Sets the logging level, specifying one of the following attributes: • `TRACE` • `NOTIFICATION` • `WARNING` • `ERROR` • `INCIDENT_ERROR` The default logging level is `NOTIFICATION`. Consider setting the `-logLevel TRACE` attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if `-logLevel TRACE` is used.

Table 4-3 (Cont.) Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-logDir` *location*	Optional	Sets the default location of upgrade log files and temporary files. You must specify an existing, writable directory where the Upgrade Assistant creates log files and temporary files. The default locations are: (UNIX) *ORACLE_HOME*`/oracle_common/upgrade/logs` *ORACLE_HOME*`/oracle_common/upgrade/temp` (Windows) *ORACLE_HOME*`\oracle_common\upgrade\logs` *ORACLE_HOME*`\oracle_common\upgrade\temp`
`-help`	Optional	Displays all of the command-line options.

Performing a Readiness Check with the Upgrade Assistant

Navigate through the screens in the Upgrade Assistant to complete the pre-upgrade readiness check.

Readiness checks are performed only on schemas or component configurations that are at a supported upgrade starting point.

To complete the readiness check:

1. On the Welcome screen, review information about the readiness check. Click **Next**.
2. On the Readiness Check Type screen, select the readiness check that you want to perform:
   - **Individually Selected Schemas** allows you to select individual schemas for review before upgrade. The readiness check reports whether a schema is supported for an upgrade or where an upgrade is needed.
     When you select this option, the screen name changes to Selected Schemas.
   - **Domain Based** allows the Upgrade Assistant to discover and select all upgrade-eligible schemas or component configurations in the domain specified in the **Domain Directory** field.
     When you select this option, the screen name changes to Schemas and Configuration.

     Leave the default selection if you want the Upgrade Assistant to check all schemas and component configurations at the same time, or select a specific option:
     - **Include checks for all schemas** to discover and review all components that have a schema available to upgrade.
     - **Include checks for all configurations** to review component configurations for a managed WebLogic Server domain.

> **Note:**
>
> If you are running an enterprise type of deployment, the domain directory will be the directory where your Administration Server runs.

Click **Next**.

3. If you selected **Individually Selected Schemas**: On the Available Components screen, select the components that have a schema available to upgrade for which you want to perform a readiness check.

   If you selected **Domain Based**: On the Component List screen, review the list of components that are present in your domain for which you want to perform a readiness check.

   If you select a component that has dependent components, those components are automatically selected. For example, if you select Oracle Platform Security Services, Oracle Audit Services is automatically selected.

   Depending on the components you select, additional screens may display. For example, you may need to:

   - Specify the Administrator server domain directory.

     Ensure that you specify the 12c (12.2.1.4.0) Administrator server domain directory.

   - Specify schema credentials to connect to the selected schema: **Database Type**, **DBA User Name**, and **DBA Password**. As part of the pre-upgrade requirements, you had created the required user, see Creating a Non-SYSDBA User to Run the Upgrade Assistant.

     Then click **Connect**.

     > **Note:**
     >
     > Oracle database is the default database type. Make sure that you select the correct database type before you continue. If you discover that you selected the wrong database type, do not go back to this screen to change it to the correct type. Instead, close the Upgrade Assistant and restart the readiness check with the correct database type selected to ensure that the correct database type is applied to all schemas.

   - Select the **Schema User Name** option and specify the **Schema Password**.

     > **Note:**
     >
     > The Upgrade Assistant automatically enables default credentials. If you are unable to connect, make sure that you manually enter the credentials for your schema before you continue.

   Click **Next** to start the readiness check.

4. On the Readiness Summary screen, review the summary of the readiness checks that will be performed based on your selections.

   If you want to save your selections to a response file to run the Upgrade Assistant again later in response (or silent) mode, click **Save Response File** and provide the location and

name of the response file. A silent upgrade performs exactly the same function that the Upgrade Assistant performs, but you do not have to manually enter the data again.

For a detailed report, click **View Log**.

Click **Next**.

5. On the Readiness Check screen, review the status of the readiness check. The process can take several minutes.

   If you are checking multiple components, the progress of each component displays in its own progress bar in parallel.

   When the readiness check is complete, click **Continue**.

   The following components are marked as **ready for upgrade** although they are not upgraded. Ignore the **ready for upgrade** message against these components:

   - Oracle JRF
   - Common Infrastructure Services
   - Oracle Web Services Manager

6. On the End of Readiness screen, review the results of the readiness check (**Readiness Success** or **Readiness Failure**):

   - If the readiness check is successful, click **View Readiness Report** to review the complete report. Oracle recommends that you review the Readiness Report before you perform the actual upgrade even when the readiness check is successful. Use the **Find** option to search for a particular word or phrase within the report. The report also indicates where the completed Readiness Check Report file is located.
   - If the readiness check encounters an issue or error, click **View Log** to review the log file, identify and correct the issues, and then restart the readiness check. The log file is managed by the command-line options you set.

Understanding the Readiness Report

After performing a readiness check for your domain, review the report to determine whether you need to take any action for a successful upgrade.

The format of the readiness report file is:

`readiness<timestamp>.txt`

Where, *timestamp* indicates the date and time of when the readiness check was run.

A readiness report contains the following information:

Table 4-4 Readiness Report Elements

Report Information	Description	Required Action
Overall Readiness Status: SUCCESS or FAILURE	The top of the report indicates whether the readiness check passed or completed with one or more errors.	If the report completed with one or more errors, search for FAIL and correct the failing issues before attempting to upgrade. You can re-run the readiness check as many times as necessary before an upgrade.
Timestamp	The date and time that the report was generated.	No action required.

Table 4-4 (Cont.) Readiness Report Elements

Report Information	Description	Required Action
Log file location `/oracle_common/upgrade/logs`	The directory location of the generated log file.	No action required.
Domain Directory	Displays the domain location	No action required.
Readiness report location `/oracle_common/upgrade/logs`	The directory location of the generated readiness report.	No action required.
Names of components that were checked	The names and versions of the components included in the check and status.	If your domain includes components that cannot be upgraded to this release, such as SOA Core Extension, do not attempt an upgrade.
Names of schemas that were checked	The names and current versions of the schemas included in the check and status.	Review the version numbers of your schemas. If your domain includes schemas that cannot be upgraded to this release, do not attempt an upgrade.
Individual Object Test Status: FAIL	The readiness check test detected an issue with a specific object.	Do not upgrade until all failed issues have been resolved.
Individual Object Test Status: PASS	The readiness check test detected no issues for the specific object.	If your readiness check report shows only the PASS status, you can upgrade your environment. Note, however, that the Readiness Check cannot detect issues with externals such as hardware or connectivity during an upgrade. You should always monitor the progress of your upgrade.
Completed Readiness Check of <Object> Status: FAILURE	The readiness check detected one or more errors that must be resolved for a particular object such as a schema, an index, or datatype.	Do not upgrade until all failed issues have been resolved.
Completed Readiness Check of <Object> Status: SUCCESS	The readiness check test detected no issues.	No action required.

Here is a sample Readiness Report file. Your report may not include all of these checks.

```
Upgrade readiness check completed with one or more errors.

This readiness check report was created on Fri Aug 16 13:29:41 PDT 2019
Log file is located at: /oracle/work/middleware_latest/oracle_common/upgrade/
logs/ua2019-08-16-13-23-36PM.log
Readiness Check Report File: /oracle/work/middleware_latest/oracle_common/
upgrade/logs/readiness2019-08-16-13-29-41PM.txt
Domain Directory: /oracle/work/middleware_1212/user_projects/domains/
jrf_domain

Starting readiness check of components.

Oracle Platform Security Services
   Starting readiness check of Oracle Platform Security Services.
     Schema User Name: DEV3_OPSS
     Database Type: Oracle Database
```

```
     Database Connect String:
     VERSION Schema DEV3_OPSS is currently at version 12.1.2.0.0.  Readiness
checks will now be performed.
   Starting schema test:  TEST_DATABASE_VERSION  Test that the database
server version number is supported for upgrade
     INFO   Database product version: Oracle Database 12c Enterprise Edition
Release 12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing
options
   Completed schema test: TEST_DATABASE_VERSION --> Test that the database
server version number is supported for upgrade +++ PASS
   Starting schema test:  TEST_REQUIRED_TABLES  Test that the schema contains
all the required tables
   Completed schema test: TEST_REQUIRED_TABLES --> Test that the schema
contains all the required tables +++ PASS
   Starting schema test:  Test that the schema does not contain any
unexpected tables  TEST_UNEXPECTED_TABLES
   Completed schema test: Test that the schema does not contain any
unexpected tables --> TEST_UNEXPECTED_TABLES +++ Test that the schema does
not contain any unexpected tables
   Starting schema test:  TEST_ENOUGH_TABLESPACE  Test that the schema
tablespaces automatically extend if full
   Completed schema test: TEST_ENOUGH_TABLESPACE --> Test that the schema
tablespaces automatically extend if full +++ PASS
   Starting schema test:  TEST_USER_TABLESPACE_QUOTA  Test that tablespace
quota for this user is sufficient to perform the upgrade
   Completed schema test: TEST_USER_TABLESPACE_QUOTA --> Test that tablespace
quota for this user is sufficient to perform the upgrade +++ PASS
   Starting schema test:  TEST_ONLINE_TABLESPACE  Test that schema
tablespaces are online
   Completed schema test: TEST_ONLINE_TABLESPACE --> Test that schema
tablespaces are online +++ PASS
   Starting permissions test:  TEST_DBA_TABLE_GRANTS  Test that DBA user has
privilege to view all user tables
   Completed permissions test: TEST_DBA_TABLE_GRANTS --> Test that DBA user
has privilege to view all user tables +++ PASS
   Starting schema test:  SEQUENCE_TEST  Test that the Oracle Platform
Security Services schema sequence and its properties are valid
   Completed schema test: SEQUENCE_TEST --> Test that the Oracle Platform
Security Services schema sequence and its properties are valid +++ PASS
   Finished readiness check of Oracle Platform Security Services with status:
SUCCESS.

Oracle Audit Services
   Starting readiness check of Oracle Audit Services.
     Schema User Name: DEV3_IAU
     Database Type: Oracle Database
     Database Connect String:
     VERSION Schema DEV3_IAU is currently at version 12.1.2.0.0.  Readiness
checks will now be performed.
   Starting schema test:  TEST_DATABASE_VERSION  Test that the database
server version number is supported for upgrade
     INFO   Database product version: Oracle Database 12c Enterprise Edition
Release 12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing
options
```

```
   Completed schema test: TEST_DATABASE_VERSION --> Test that the database
server version number is supported for upgrade +++ PASS
   Starting schema test:  TEST_REQUIRED_TABLES  Test that the schema contains
all the required tables
   Completed schema test: TEST_REQUIRED_TABLES --> Test that the schema
contains all the required tables +++ PASS
   Starting schema test:  TEST_UNEXPECTED_TABLES  Test that the schema does
not contain any unexpected tables
   Completed schema test: TEST_UNEXPECTED_TABLES --> Test that the schema
does not contain any unexpected tables +++ PASS
   Starting schema test:  TEST_ENOUGH_TABLESPACE  Test that the schema
tablespaces automatically extend if full
   Completed schema test: TEST_ENOUGH_TABLESPACE --> Test that the schema
tablespaces automatically extend if full +++ PASS
   Starting schema test:  TEST_USER_TABLESPACE_QUOTA  Test that tablespace
quota for this user is sufficient to perform the upgrade
   Completed schema test: TEST_USER_TABLESPACE_QUOTA --> Test that tablespace
quota for this user is sufficient to perform the upgrade +++ PASS
   Starting schema test:  TEST_ONLINE_TABLESPACE  Test that schema
tablespaces are online
   Completed schema test: TEST_ONLINE_TABLESPACE --> Test that schema
tablespaces are online +++ PASS
   Starting permissions test:  TEST_DBA_TABLE_GRANTS  Test that DBA user has
privilege to view all user tables
   Completed permissions test: TEST_DBA_TABLE_GRANTS --> Test that DBA user
has privilege to view all user tables +++ PASS
   Starting schema test:  TEST_MISSING_COLUMNS  Test that tables and views
are not missing any required columns
   Completed schema test: TEST_MISSING_COLUMNS --> Test that tables and views
are not missing any required columns +++ PASS
   Starting schema test:  TEST_UNEXPECTED_COLUMNS  Test that tables and views
do not contain any unexpected columns
   Completed schema test: TEST_UNEXPECTED_COLUMNS --> Test that tables and
views do not contain any unexpected columns +++ PASS
   Starting datatype test for table OIDCOMPONENT:  TEST_COLUMN_DATATYPES_V2 --
> Test that all table columns have the proper datatypes
   Completed datatype test for table OIDCOMPONENT: TEST_COLUMN_DATATYPES_V2 --
> Test that all table columns have the proper datatypes +++ PASS
   Starting datatype test for table IAU_CUSTOM_01:  TEST_COLUMN_DATATYPES_V2
--> Test that all table columns have the proper datatypes
   Completed datatype test for table IAU_CUSTOM_01: TEST_COLUMN_DATATYPES_V2
--> Test that all table columns have the proper datatypes +++ PASS
   Starting datatype test for table IAU_BASE:  TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes
   Completed datatype test for table IAU_BASE: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes +++ PASS
   Starting datatype test for table WS_POLICYATTACHMENT:
TEST_COLUMN_DATATYPES_V2 --> Test that all table columns have the proper
datatypes
   Completed datatype test for table WS_POLICYATTACHMENT:
TEST_COLUMN_DATATYPES_V2 --> Test that all table columns have the proper
datatypes +++ PASS
   Starting datatype test for table OWSM_PM_EJB:  TEST_COLUMN_DATATYPES_V2 --
> Test that all table columns have the proper datatypes
   Completed datatype test for table OWSM_PM_EJB: TEST_COLUMN_DATATYPES_V2 --
> Test that all table columns have the proper datatypes +++ PASS
```

```
   Starting datatype test for table XMLPSERVER:  TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes
   Completed datatype test for table XMLPSERVER: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes +++ PASS
   Starting datatype test for table SOA_HCFP:  TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes
   Completed datatype test for table SOA_HCFP: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes +++ PASS
   Starting schema test:  SEQUENCE_TEST  Test that the audit schema sequence
and its properties are valid
   Completed schema test: SEQUENCE_TEST --> Test that the audit schema
sequence and its properties are valid +++ PASS
   Starting schema test:  SYNONYMS_TEST  Test that the audit schema required
synonyms are present
   Completed schema test: SYNONYMS_TEST --> Test that the audit schema
required synonyms are present +++ PASS
   Finished readiness check of Oracle Audit Services with status: FAILURE.

Common Infrastructure Services
   Starting readiness check of Common Infrastructure Services.
     Schema User Name: DEV3_STB
     Database Type: Oracle Database
     Database Connect String:
   Starting schema test:  TEST_REQUIRED_TABLES  Test that the schema contains
all the required tables
   Completed schema test: TEST_REQUIRED_TABLES --> Test that the schema
contains all the required tables +++ PASS
   Completed schema test: ALL_TABLES --> TEST_REQUIRED_TABLES +++ Test that
the schema contains all the required tables
   Starting schema test:  TEST_UNEXPECTED_TABLES  Test that the schema does
not contain any unexpected tables
   Completed schema test: ALL_TABLES --> TEST_UNEXPECTED_TABLES +++ Test that
the schema does not contain any unexpected tables
   Starting schema test:  TEST_REQUIRED_VIEWS  Test that the schema contains
all the required database views
   Completed schema test: ALL_TABLES --> TEST_REQUIRED_VIEWS +++ Test that
the schema contains all the required database views
   Starting schema test:  TEST_MISSING_COLUMNS  Test that tables and views
are not missing any required columns
   Completed schema test: ALL_TABLES --> TEST_MISSING_COLUMNS +++ Test that
tables and views are not missing any required columns
   Starting schema test:  TEST_DATABASE_VERSION  Test that the database
server version number is supported for upgrade
   Starting schema test:  TEST_DATABASE_VERSION  Test that the database
server version number is supported for upgrade
     INFO   Database product version: Oracle Database 12c Enterprise Edition
Release 12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing
options
   Completed schema test: TEST_DATABASE_VERSION --> Test that the database
server version number is supported for upgrade +++ PASS
   Completed schema test: ALL_TABLES --> TEST_DATABASE_VERSION +++ Test that
the database server version number is supported for upgrade
   Finished readiness check of Common Infrastructure Services with status:
SUCCESS.
```

```
Oracle JRF
   Starting readiness check of Oracle JRF.
   Finished readiness check of Oracle JRF with status: SUCCESS.

System Components Infrastructure
   Starting readiness check of System Components Infrastructure.
   Starting config test:  TEST_SOURCE_CONFIG  Checking the source
configuration.
     INFO /oracle/work/middleware_1212/user_projects/domains/jrf_domain/opmn/
topology.xml was not found. No upgrade is needed.
   Completed config test: TEST_SOURCE_CONFIG --> Checking the source
configuration. +++ PASS
   Finished readiness check of System Components Infrastructure with status:
ALREADY_UPGRADED.

Common Infrastructure Services
   Starting readiness check of Common Infrastructure Services.
   Starting config test:  CIEConfigPlugin.readiness.test  This tests the
readiness of the domain from CIE side.
   Completed config test: CIEConfigPlugin.readiness.test --> This tests the
readiness of the domain from CIE side. +++ PASS
   Finished readiness check of Common Infrastructure Services with status:
SUCCESS.

Finished readiness check of components.
```

Upgrading Product Schemas From OIMHOST1

Upgrade all of the necessary schemas for Oracle Identity Manager, from OIMHOST1 by using the Upgrade Assistant.

- Upgrading Product Schemas
  After stopping servers and processes, use the Upgrade Assistant to upgrade supported product schemas to the current release of Oracle Fusion Middleware.

Upgrading Product Schemas

After stopping servers and processes, use the Upgrade Assistant to upgrade supported product schemas to the current release of Oracle Fusion Middleware.

The Upgrade Assistant allows you to upgrade individually selected schemas or all schemas associated with a domain. The option you select determines which Upgrade Assistant screens you will use.

Note:

High waits and performance degradation may be seen due to 'library cache lock' (cycle)<='library cache lock' for DataPump Worker (DW) processes in the 12.2 RAC environment. To resolve this issue, you should disable S-Optimization by using the following command:

```
ALTER SYSTEM SET "_lm_share_lock_opt"=FALSE SCOPE=SPFILE SID='*';
```

After running the above command, restart all the RAC instances. After the upgrade is complete, you can reset the parameter by using the following command:

```
alter system reset "_lm_share_lock_opt" scope=spfile sid='*';
```

- Identifying Existing Schemas Available for Upgrade
  This optional step can be used before an upgrade to query the schema version registry table. This table contains schema information such as the schema owner, version number, component name and ID, date of creation and modification, and custom prefixes.
- Starting the Upgrade Assistant
  Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 14c (14.1.2.1.0).
- Upgrading Oracle Identity Manager Schemas Using the Upgrade Assistant
  Navigate through the screens in the Upgrade Assistant to upgrade the product schemas.
- Verifying the Schema Upgrade
  After completing all the upgrade steps, verify that the upgrade was successful by checking that the schema version in `schema_version_registry` has been properly updated.

Identifying Existing Schemas Available for Upgrade

This optional step can be used before an upgrade to query the schema version registry table. This table contains schema information such as the schema owner, version number, component name and ID, date of creation and modification, and custom prefixes.

You can let the Upgrade Assistant upgrade all of the schemas in the domain, or you can select individual schemas to upgrade. To help decide, follow these steps to view a list of all the schemas that are available for an upgrade:

1. If you are using an Oracle database, connect to the database by using an account that has Oracle DBA privileges, and run the following from SQL*Plus:

```
SET LINE 120
COLUMN MRC_NAME FORMAT A14
COLUMN COMP_ID FORMAT A20
COLUMN VERSION FORMAT A12
COLUMN STATUS FORMAT A9
COLUMN UPGRADED FORMAT A8
SELECT MRC_NAME, COMP_ID, OWNER, VERSION, STATUS, UPGRADED FROM
SCHEMA_VERSION_REGISTRY WHERE OWNER LIKE UPPER('<PREFIX>_%');
```

2. Examine the report that is generated.

Notes:

- After the upgrade you can generate the report again to see the updated versions of your schemas. If an upgrade was not needed for a schema, the `schema_version_registry` table retains the schema at its pre-upgrade version.
- If your existing schemas are not from a supported version, then you must upgrade them to a supported version before using the 14c (14.1.2.1.0) upgrade procedures. Refer to your pre-upgrade version documentation for more information.
- If you used an OID-based policy store in the earlier versions, make sure to create a new OPSS schema before you perform the upgrade. After the upgrade, the OPSS schema remains an LDAP-based store.
- You can only upgrade schemas for products that are available for upgrade in Oracle Fusion Middleware release 14c (14.1.2.1.0). Do not attempt to upgrade a domain that includes components that are not yet available for upgrade to 14c (14.1.2.1.0).

Starting the Upgrade Assistant

Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 14c (14.1.2.1.0).

To start the Upgrade Assistant:

Note:

Before you start the Upgrade Assistant, make sure that the JVM character encoding is set to UTF-8 for the platform on which the Upgrade Assistant is running. If the character encoding is not set to UTF-8, then you will not be able to download files containing Unicode characters in their names. This can cause the upgrade to fail. To set the character encoding, run the following:

UNIX operating systems:

```
export UA_PROPERTIES="-Dfile.encoding=UTF-8 ${UA_PROPERTIES}"
```

Windows operating systems:

```
set UA_PROPERTIES=-Dfile.encoding=UTF-8 %UA_PROPERTIES%
```

1. Go to the `oracle_common/upgrade/bin` directory:
   - (UNIX) *ORACLE_HOME*`/oracle_common/upgrade/bin`
   - (Windows) *ORACLE_HOME*`\oracle_common\upgrade\bin`
2. Start the Upgrade Assistant:
   - (UNIX) `./ua`

- (Windows) `ua.bat`

For information about other parameters that you can specify on the command line, such as logging parameters, see:

- Upgrade Assistant Parameters

Upgrade Assistant Parameters

When you start the Upgrade Assistant from the command line, you can specify additional parameters.

Table 4-5 Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-readiness`	Required for readiness checks **Note**: Readiness checks cannot be performed on standalone installations (those not managed by the WebLogic Server).	Performs the upgrade readiness check without performing an actual upgrade. Schemas and configurations are checked. Do not use this parameter if you have specified the `-examine` parameter.
`-threads`	Optional	Identifies the number of threads available for concurrent schema upgrades or readiness checks of the schemas. The value must be a positive integer in the range 1 to 8. The default is 4.
`-response`	Required for silent upgrades or silent readiness checks	Runs the Upgrade Assistant using inputs saved to a response file generated from the data that is entered when the Upgrade Assistant is run in GUI mode. Using this parameter runs the Upgrade Assistant in *silent mode* (without displaying Upgrade Assistant screens).
`-examine`	Optional	Performs the examine phase but does not perform an actual upgrade. Do not specify this parameter if you have specified the `-readiness` parameter.

Table 4-5 (Cont.) Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-logLevel` *attribute*	Optional	Sets the logging level, specifying one of the following attributes: • `TRACE` • `NOTIFICATION` • `WARNING` • `ERROR` • `INCIDENT_ERROR` The default logging level is `NOTIFICATION`. Consider setting the `-logLevel TRACE` attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if `-logLevel TRACE` is used.
`-logDir` *location*	Optional	Sets the default location of upgrade log files and temporary files. You must specify an existing, writable directory where the Upgrade Assistant creates log files and temporary files. The default locations are: (UNIX) *ORACLE_HOME*`/oracle_common/upgrade/logs` *ORACLE_HOME*`/oracle_common/upgrade/temp` (Windows) *ORACLE_HOME*`\oracle_common\upgrade\logs` *ORACLE_HOME*`\oracle_common\upgrade\temp`
`-help`	Optional	Displays all of the command-line options.

Upgrading Oracle Identity Manager Schemas Using the Upgrade Assistant

Navigate through the screens in the Upgrade Assistant to upgrade the product schemas.

To upgrade product schemas with the Upgrade Assistant:

1. On the Welcome screen, review an introduction to the Upgrade Assistant and information about important pre-upgrade tasks. Click **Next**.

> **Note:**
>
> For more information about any Upgrade Assistant screen, click **Help** on the screen.

2. On the Upgrade Type screen, select the schema upgrade operation that you want to perform:
   - **Individually Selected Schemas** if you want to select individual schemas for upgrade and you do not want to upgrade all of the schemas used by the domain.

     > **Caution:**
     >
     > Upgrade only those schemas that are used to support your 14c (14.1.2.1.0) components. Do not upgrade schemas that are currently being used to support components that are not included in Oracle Fusion Middleware 14c (14.1.2.1.0).

   - **All Schemas Used by a Domain** to allow the Upgrade Assistant to discover and select all components that have a schema available to upgrade in the domain specified in the **Domain Directory** field. This is also known as a *domain assisted schema upgrade*. Additionally, the Upgrade Assistant pre-populates connection information on the schema input screens.

     > **Note:**
     >
     > Oracle recommends that you select **All Schemas Used by a Domain** for most upgrades to ensure all of the required schemas are included in the upgrade.

   > **Note:**
   >
   > If you are upgrading SSL enabled Oracle Identity Manager setup, select **Individually Selected Schemas** option, and then select Oracle Identity Manager schema only. This automatically selects the dependant schemas. For upgrading SSL enabled setup, you must provide the non-SSL Database connection details on the Schema Credentials screen.

3. If you selected **Individually Selected Schemas**: On the Available Components screen, select the components for which you want to upgrade schemas. When you select a component, the schemas and any dependencies are automatically selected.

> **Note:**
>
> - For the individual schema option, the domain configuration is not accessed, and therefore password values are carried forward from the previous screen. If you encounter any connection failure, check the cause and fix it.
> - For the Upgrade Assistant utility to use the correct UMS schema, manually edit the UMS schema by adding `_UMS` as a suffix. For example, edit `DEV` to `DEV_UMS` for successful SOA upgrade.

4. On the Screen name, select the domain folder.

   Click **Next**.

5. On the Component List screen, it will display the list of components whose schema will be upgraded.

   Click **Next**.

6. On the Prerequisites screen, acknowledge that the prerequisites have been met by selecting all the check boxes. Click **Next**.

> **Note:**
>
> The Upgrade Assistant does not verify whether the prerequisites have been met.

7. On the Schema Credentials screen(s), specify the database connection details for each schema you are upgrading (the screen name changes based on the schema selected):
   - Select the database type from the **Database Type** drop-down menu.
   - Enter the database connection details, and click **Connect**.
   - Select the schema you want to upgrade from the **Schema User Name** drop-down menu, and then enter the password for the schema. Be sure to use the correct schema prefix for the schemas you are upgrading.

   Click **Next**.

8. On the Examine screen, review the status of the Upgrade Assistant as it examines each schema, verifying that the schema is ready for upgrade. If the status is **Examine finished**, click **Next**.

   If the examine phase fails, Oracle recommends that you cancel the upgrade by clicking **No** in the Examination Failure dialog. Click **View Log** to see what caused the error and refer to Troubleshooting Your Upgrade in *Upgrading with the Upgrade Assistant* for information on resolving common upgrade errors.

> **Note:**
>
> - If you resolve any issues detected during the examine phase without proceeding with the upgrade, you can start the Upgrade Assistant again without restoring from backup. However, if you proceed by clicking **Yes** in the Examination Failure dialog box, you need to restore your pre-upgrade environment from backup before starting the Upgrade Assistant again.
> - Canceling the examination process has no effect on the schemas or configuration data; the only consequence is that the information the Upgrade Assistant has collected must be collected again in a future upgrade session.

9. On the Upgrade Summary screen, review the summary of the options you have selected for schema upgrade.

   Verify that the correct Source and Target Versions are listed for each schema you intend to upgrade.

   If you want to save these options to a response file to run the Upgrade Assistant again later in response (or silent) mode, click **Save Response File** and provide the location and name of the response file. A silent upgrade performs exactly the same function that the Upgrade Assistant performs, but you do not have to manually enter the data again.

   Click **Upgrade** to start the upgrade process.

10. On the Upgrade Progress screen, monitor the status of the upgrade.

> **Caution:**
>
> Allow the Upgrade Assistant enough time to perform the upgrade. Do not cancel the upgrade operation unless absolutely necessary. Doing so may result in an unstable environment.

   If any schemas are not upgraded successfully, refer to the Upgrade Assistant log files for more information.

> **Note:**
>
> The progress bar on this screen displays the progress of the current upgrade procedure. It does not indicate the time remaining for the upgrade.

   Click **Next**.

11. After the upgrade completes successfully, the Upgrade Assistant provides the upgrade status and lists the next steps to take in the upgrade process. You should review the Upgrade Success screen of the Upgrade Assistant to determine the next steps based on the information provided. The wizard shows the following information:

```
Upgrade Succeeded.

Log File: /u01/oracle/products/12c/identity/oracle_common/upgrade/logs/
ua2020-09-15-18-27-29PM.txt
Post Upgrade Text file: /u01/oracle/products/12c/identity/oracle_common/upgrade/logs/
postupgrade2020-09-15-18-27-29PM.txt
```

```
Next Steps

Oracle SOA
1. The Upgrade Assistant has successfully upgraded all active instances. You can now
close the Upgrade Assistant.
2. The automated upgrade of closed instances will continue in the background after
the Upgrade Assistant is exited and until the SOA server is started,at which point
the upgrade will stop. You can schedule the upgrade of any remaining closed
instances for a time when the SOA server is less busy.
   Close the Upgrade Assistant and use the instance data administration scripts to
administer and monitor the overall progress of this automated upgrade. For more
information see "Administering and Monitoring the Upgrade of SOA Instance Data" in
Upgrading SOA Suite and Business Process Management.
```

Click **Close** to complete the upgrade and close the wizard.

If the upgrade fails: On the Upgrade Failure screen, click **View Log** to view and troubleshoot the errors. The logs are available at *ORACLE_HOME*/oracle_common/upgrade/logs.

> **Note:**
>
> If the upgrade fails, you must restore your pre-upgrade environment from backup, fix the issues, then restart the Upgrade Assistant.

Verifying the Schema Upgrade

After completing all the upgrade steps, verify that the upgrade was successful by checking that the schema version in `schema_version_registry` has been properly updated.

If you are using an Oracle database, connect to the database as a user having Oracle DBA privileges, and run the following from SQL*Plus to get the current version numbers. Be sure to replace <*PREFIX*> with your schema prefix.

```
SET LINE 120
COLUMN MRC_NAME FORMAT A14
COLUMN COMP_ID FORMAT A20
COLUMN VERSION FORMAT A12
COLUMN STATUS FORMAT A9
COLUMN UPGRADED FORMAT A8
SELECT MRC_NAME, COMP_ID, OWNER, EDITION NAME, VERSION, STATUS, UPGRADED FROM
SCHEMA_VERSION_REGISTRY where owner like '<PREFIX>_%';
```

In the query result:

- Verify that the `EDITION NAME` column appears as `ORA$BASE`.
- Check that the number in the `VERSION` column matches the latest version number for that schema. For example, verify that the schema version number is 14.1.2.1.0.

> **Note:**
>
> Not all schema versions will be updated. Some schemas do not require an upgrade to this release and will retain their pre-upgrade version number.

- The `STATUS` field will be either `UPGRADING` or `UPGRADED` during the schema patching operation, and will become `VALID` when the operation is completed.
- If the status appears as `INVALID`, the schema update failed. You should examine the logs files to determine the reason for the failure.
- Synonym objects owned by `IAU_APPEND` and `IAU_VIEWER` will appear as `INVALID`, but that does not indicate a failure.

  They become invalid because the target object changes after the creation of the synonym. The synonyms objects will become valid when they are accessed. You can safely ignore these `INVALID` objects.

Upgrading Domain Component Configurations on OIMHOST1

Use the Upgrade Assistant to upgrade the domain component's configurations inside the domain to match the updated domain configuration.

> **Note:**
>
> Perform this procedure on OIMHOST1 only.

- Upgrading Domain Component Configurations
  Use the Upgrade Assistant to upgrade the domain *component* configurations inside the domain to match the updated domain configuration.

Upgrading Domain Component Configurations

Use the Upgrade Assistant to upgrade the domain *component* configurations inside the domain to match the updated domain configuration.

- Starting the Upgrade Assistant
  Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 14c (14.1.2.1.0).
- Upgrading Oracle Identity Manager Domain Component Configurations
  Navigate through the screens in the Upgrade Assistant to upgrade component configurations in the WebLogic domain.

Starting the Upgrade Assistant

Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 14c (14.1.2.1.0).

To start the Upgrade Assistant:

> **Note:**
>
> Before you start the Upgrade Assistant, make sure that the JVM character encoding is set to UTF-8 for the platform on which the Upgrade Assistant is running. If the character encoding is not set to UTF-8, then you will not be able to download files containing Unicode characters in their names. This can cause the upgrade to fail. To set the character encoding, run the following:
>
> UNIX operating systems:
>
> ```
> export UA_PROPERTIES="-Dfile.encoding=UTF-8 ${UA_PROPERTIES}"
> ```
>
> Windows operating systems:
>
> ```
> set UA_PROPERTIES=-Dfile.encoding=UTF-8 %UA_PROPERTIES%
> ```

1. Go to the `oracle_common/upgrade/bin` directory:
   - (UNIX) *`ORACLE_HOME`*`/oracle_common/upgrade/bin`
   - (Windows) *`ORACLE_HOME`*`\oracle_common\upgrade\bin`
2. Start the Upgrade Assistant:
   - (UNIX) `./ua`
   - (Windows) `ua.bat`

For information about other parameters that you can specify on the command line, such as logging parameters, see:

Upgrading Oracle Identity Manager Domain Component Configurations

Navigate through the screens in the Upgrade Assistant to upgrade component configurations in the WebLogic domain.

Run the Upgrade Assistant to upgrade the domain component configurations to match the updated domain configuration.

To upgrade domain component configurations with the Upgrade Assistant:

1. On the Welcome screen, review an introduction to the Upgrade Assistant and information about important pre-upgrade tasks. Click **Next**.

   > **Note:**
   >
   > For more information about any Upgrade Assistant screen, click **Help** on the screen.

2. On the next screen:
   - Select **All Configurations Used By a Domain**. The screen name changes to WebLogic Components.
   - In the **Domain Directory** field, specify the OIM domain directory.

Where, **Domain Directory** is the Administration server domain directory.

Click **Next**.

3. On the Component List screen, verify that the list includes all the components for which you want to upgrade configurations and click **Next**.

   If you do not see the components you want to upgrade, click **Back** to go to the previous screen and specify a different domain.

4. On the Prerequisites screen, acknowledge that the prerequisites have been met by selecting all the check boxes. Click **Next**.

   > **Note:**
   >
   > The Upgrade Assistant does not verify whether the prerequisites have been met.

5. On the Examine screen, review the status of the Upgrade Assistant as it examines each component, verifying that the component configuration is ready for upgrade. If the status is **Examine finished**, click **Next**.

   If the examine phase fails, Oracle recommends that you cancel the upgrade by clicking **No** in the Examination Failure dialog. Click **View Log** to see what caused the error and refer to Troubleshooting Your Upgrade in *Upgrading with the Upgrade Assistant* for information on resolving common upgrade errors.

   > **Note:**
   >
   > - If you resolve any issues detected during the examine phase without proceeding with the upgrade, you can start the Upgrade Assistant again without restoring from backup. However, if you proceed by clicking **Yes** in the Examination Failure dialog box, you need to restore your pre-upgrade environment from backup before starting the Upgrade Assistant again.
   > - Canceling the examination process has no effect on the configuration data; the only consequence is that the information the Upgrade Assistant has collected must be collected again in a future upgrade session.

6. On the Upgrade Summary screen, review the summary of the options you have selected for component configuration upgrade.

   The response file collects and stores all the information that you have entered, and enables you to perform a silent upgrade at a later time. The silent upgrade performs exactly the same function that the Upgrade Assistant performs, but you do not have to manually enter the data again. If you want to save these options to a response file, click **Save Response File** and provide the location and name of the response file.

   Click **Upgrade** to start the upgrade process.

7. On the Upgrade Progress screen, monitor the status of the upgrade.

> **Caution:**
>
> Allow the Upgrade Assistant enough time to perform the upgrade. Do not cancel the upgrade operation unless absolutely necessary. Doing so may result in an unstable environment.

If any components are not upgraded successfully, refer to the Upgrade Assistant log files for more information.

> **Note:**
>
> The progress bar on this screen displays the progress of the current upgrade procedure. It does not indicate the time remaining for the upgrade.

Click **Next**.

8. If the upgrade is successful: On the Upgrade Success screen, click **Close** to complete the upgrade and close the wizard. The Post-Upgrade Actions window describes the manual tasks you must perform to make components functional in the new installation. This window appears only if a component has post-upgrade steps.

   If the upgrade fails: On the Upgrade Failure screen, click **View Log** to view and troubleshoot the errors. The logs are available at *ORACLE_HOME*/oracle_common/upgrade/logs.

> **Note:**
>
> If the upgrade fails you must restore your pre-upgrade environment from backup, fix the issues, then restart the Upgrade Assistant.

Verifying the Domain-Specific-Component Configurations Upgrade

To verify that the domain-specific-component configurations upgrade was successful, sign in to the Administration console and the Oracle Enterprise Manager Fusion Middleware Control and verify that the version numbers for each component is 14.1.2.1.0.

To sign in to the Administration Console, go to: http://*administration_server_host*:*administration_server_port*/console

To sign in to Oracle Enterprise Manager Fusion Middleware Control Console, go to: http://*administration_server_host*:*administration_server_port*/em

Updating the setDomainEnv.sh File

For upgrading Oracle Identity Governance (OIG) from 12c (12.2.1.4.0) to 14c (14.1.2.1.0), you need to the delete a property in the setDomainEnv.sh file.

Complete the following steps:

1. Open the `setDomainEnv.sh` file in the *`Oracle_Home`*`/domains/<domain name>/bin/` location.
2. Delete the following parameter from the line which starts as follows:

   ```
   EXTRA_JAVA_PROPERTIES="-Djavax.net.ssl.trustStore=${WL_HOME}/server/lib/
   DemoTrust.jks
   ```

   The parameter is:

   ```
   -Doracle.xdkjava.compatibility.version=11.1.1
   ```

3. Save and close the `setDomainEnv.sh` file.

> **Note:**
>
> - For SOA, you need to add the following entry as an argument to the `setSOADomainEnv.sh` file in the line starting with `EXTRA_JAVA_PROPERTIES="${EXTRA_JAVA_PROPERTIES}`.
>
>   ```
>   -Doracle.xdkjava.compatibility.version=11.1.1
>   ```
>
> - Repeat these steps in all the OIM host machines.

Performing OIM Bootstrap on OIMHOST1

After you upgrade Oracle Identity Manager on OIMHOST1, restart the servers.

> **Note:**
>
> If you are using an enterprise deployment where Administration and Managed servers are in different directories, restart the severs from the Administration Server directory to allow the bootstrap process to complete.

You must restart the servers in the following order:

1. Start the Administration Server. If Node manager is configured, do not start the Node Manager.
2. Start the Oracle SOA Suite Managed Server with the Administration Server URL. For example:

   ```
   ./startManagedWebLogic.sh <soa_managed_server_name> t3://
   weblogic_admin_host:weblogic_admin_port
   ```

   > **Note:**
   >
   > In an SSL environment, when you start the managed servers for the first time for bootstrap, provide the non-SSL port number of the Administration Server.

3. After the SOA server is in the running state and the **soa-infra** application in the `ACTIVE` status, start the Oracle Identity Manager Managed Server with the Administration Server URL. For example:

```
/startManagedWebLogic.sh <oim_managed_server_name> t3://
weblogic_admin_host:weblogic_admin_port
```

> **Note:**
>
> - As done in step 2, provide the non-SSL port number of the Administration Server.
> - The OIM managed server calls the **soa-infra** application when executing the bootstrap tasks. If the **soa-infra** application is not in `ACTIVE` status, then OIM bootstrap fails with the following error:
>
> ```
> <Error> <oracle.iam.OIMPostConfigManager> <BEA-000000>
> <Shutting down the
> BootStrap Process. Please fix the problem and start the OIM
> Managed server
> again to complete OIM BootStrap. OR, If you want to skip the
> feature which
> has failed, mark the feature as complete using sql 'update
> oimbootstate set
> state='COMPLETE' where featurename='FAILED_FEATURE_NAME' and
> start the
> Managed Server again. In the latter case, you will have to
> manually perform
> the task being done by the failed feature. Refer to the Install
> documentations for the same>
> java.lang.RuntimeException: None of the SOA servers are in
> RUNNING state!
> at
> oracle.iam.platform.mbeans.impl.util.SOAIntegrationUtil.getSOAS
> erverURLs(SOAIn
> tegrationUtil.java:358)
> at
> oracle.iam.OIMPostConfigManager.config.OIMConfigManager.updateO
> IMCONFIGXML(OIM
> ConfigManager.java:2939)
> ```

   After the upgrade, when the OIM server starts for the first time, the 14c (14.1.2.1.0) bootstrap starts automatically and the server is not shut down.

For more information about stopping the servers and processes, see Stopping Servers and Processes.

Handling Custom Applications

If custom applications and libraries are present in your deployment of Oracle Identity Governance (OIG) 12c (12.2.1.4.0), Oracle recommends you to update them manually after the upgrade to Oracle Identity Governance (OIG) 14c (14.1.2.1.0).

Packing Domain Configurations on OIMHOST1

After upgrading domain component configurations on OIMHOST1, pack the upgraded domain on OIMHOST1. You must unpack it later on OIMHOST2.

To do this, complete the following steps:

1. On OIMHOST1, run the following command from the location `$ORACLE_HOME/oracle_common/common/bin` to pack the upgraded domain:
   - On UNIX:
     ```
     sh pack.sh -domain=<Location_of_OIM_domain> -
     template=<Location_where_domain_configuration_jar_to_be_created> -
     template_name="OIM Domain" -managed=true
     ```
   - On Windows:
     ```
     pack.cmd -domain=<Location_of_OIM_domain> -
     template=<Location_where_domain_configuration_jar_to_be_created> -
     template_name="OIM Domain" -managed=true
     ```
2. Copy the domain configuration jar file created by the pack command on OIMHOST1 to any accessible location.

> **Note:**
>
> If you are upgrading an enterprise deployment, you need to extract the configuration to the Managed Server directory. See Replicating the Domain Configurations on Each OIMHOST.

Starting Servers and Processes

After a successful upgrade, shut down any servers you may have started manually, and then restart all processes and servers, including the Administration Server and any Managed Servers.

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

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

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

Step 1: Start Node Manager

Start the Node Manager from the Administration Server `<DOMAIN_HOME>/bin` location:

- (UNIX) `nohup ./startNodeManager.sh > <DOMAIN_HOME>/nodemanager/nodemanager.out 2>&1 &`

- (Windows) `nohup .\startNodeManager.sh > <DOMAIN_HOME>\nodemanager\nodemanager.out 2>&1 &`

Where *<DOMAIN_HOME>* is the Administration Server domain home.

Step 2: Start the Administration Server

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

If you are not using nodemanager to start Administration Server, use the `startWebLogic` script:

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

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

Step 3: Start the Managed Servers

Option 1

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

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

When prompted, enter your user name and password.

> **Note:**
>
> In an HA environment, it is preferred to use the console or node manager to start servers.

Option 2

Start a WebLogic Server Managed Server by using the Weblogic Console:

- Log into Weblogic console as a `weblogic` Admin.
- Go to **Servers** > **Control** tab.
- Select the required managed server.
- Click **Start**.

> **Note:**
>
> - The startup of a Managed Server will typically start the applications that are deployed to it. Therefore, it should not be necessary to manually start applications after the Managed Server startup.
> - The Mobile Security Manager (MSM) servers are not supported in 12c. After restarting the servers, the 11*g* configurations of MSM servers, like `omsm_server1` or `WLS_MSM1`, might remain. Ignore these configurations and do not restart the MSM servers.

Step 4: Start System Components

If required, start system components, such as Oracle HTTP Server by using the `startComponent` script:

- (UNIX) *`OHS_INSTANCE_HOME`*`/bin/startComponent.sh ohs1`
- (Windows) *`OHS_INSTANCE_HOME`*`\bin\startComponent.sh ohs1`

You can start system components in any order.

Stopping Servers and Processes on OIMHOST2

Before you upgrade the schemas and configurations, you must shut down all of the pre-upgrade processes and servers, including the Administration Server, Node Manager, and any managed servers on OIMHOST2.

An Oracle Fusion Middleware environment can consist of an Oracle WebLogic Server domain, an Administration Server, multiple managed servers, Java components, system components such as Identity Management components, and a database used as a repository for metadata. The components may be dependent on each other, so they must be stopped in the correct order.

Follow the same process that you used to stop the servers and processes on OIMHOST1. See Stopping Servers and Processes on OIMHOST1.

Upgrading the Binaries on OIMHOST2

You have to perform these steps only if OIMHOST2 is using a different binary location as compared to that of OIMHOST1.

- Uninstalling the Software on OIMHOST2
  Use the Uninstall Wizard to remove the software from the existing *ORACLE_HOME*. You will reinstall the new software into this same directory.
- Installing Product Distributions on OIMHOST2
  After you have uninstalled the software from the 12c (12.2.1.4.0) Oracle home, install the 14c (14.1.2.1.0) binaries into the same Oracle home.

Uninstalling the Software on OIMHOST2

Use the Uninstall Wizard to remove the software from the existing *ORACLE_HOME*. You will reinstall the new software into this same directory.

> **Note:**
>
> This step is necessary only if OIMHOST2 is using a different set of binaries than OIMHOST1.

Follow the same process that you used to uninstall the software on OIMHOST1. See Uninstalling the Software on OIMHOST1.
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*.

Installing Product Distributions on OIMHOST2

After you have uninstalled the software from the 12c (12.2.1.4.0) Oracle home, install the 14c (14.1.2.1.0) binaries into the same Oracle home.

Install the following products on OIMHOST2:

- Oracle Fusion Middleware Infrastructure 14*c* (14.1.2.0.0)
- Oracle SOA Suite 14*c* (14.1.2.0.0)
- Oracle Identity Manager 14c (14.1.2.1.0)

Follow the same process that you used to install the software on OIMHOST1. See Installing Product Distributions.

> **Note:**
>
> If you have redundant *Oracle_Home* installations, then install the binaries into each of the redundant locations.

Replicating the Domain Configurations on Each OIMHOST

Replicate the domain configurations on OIMHOST2. This involves unpacking the upgraded domain on OIMHOST2, which was packed on OIMHOST1.

To do this, complete the following steps:

1. Earlier in the procedure, you created a copy of the domain configuration jar file by using the pack command on OIMHOST1. See Packing Domain Configurations on OIMHOST1.

   Copy the domain configuration jar file created by the pack command on OIMHOST1 to any accessible location on OIMHOST2.
2. On OIMHOST2, rename the existing domain home to `<domain_home>_old`.
3. On OIMHOST2, run the following command from the location `$ORACLE_HOME/oracle_common/common/bin` to unpack the domain:

- On UNIX:
  ```
  sh unpack.sh -domain=<Location_of_OIM_domain> -
  template=<Location_where_domain_configuration_jar_to_be_created> -
  overwrite_domain=true
  ```
- On Windows:
  ```
  unpack.cmd -domain=<Location_of_OIM_domain> -
  template=<Location_where_domain_configuration_jar_to_be_created> -
  overwrite_domain=true
  ```

4. If you have other OIMHOSTs, repeat step 2 through step 3 on those hosts.

> **Note:**
>
> If you are following the EDG methodology, you also need to pack and unpack the domain in the OIM managed server location on OIMHOST1.

Deploy the `oracle.iam.ui.custom-dev-starter-pack.war`

Validate that the Upgrade Assistant has automatically copied the `oracle.iam.ui.custom-dev-starter-pack.war` file from the 12c (12.2.1.4.0) *MW_HOME* to the 14c (14.1.2.1.0) *ORACLE_HOME* on the AdminServer host.

If you have an Enterprise Reference topology or use multiple shared volumes for your *ORACLE_HOME* binaries, then also replicate this file manually to each OIMHOSTn where a distinct separate binary volume is mounted.

1. Check the 12c (12.2.1.4.0) *MW_HOME* for the war file, validate it is no longer present.

   ```
   ls /u01/oracle/products/identity/iam/server/apps/oracle.iam.ui.custom-dev-
   starter-pack.war
   ```

2. Check the 14*c ORACLE_HOME* for the war file, validate it has been placed in the correct location.

   ```
   ls /u01/oracle/products/12c/identity/idm/server/apps/oracle.iam.ui.custom-
   dev-starter-pack.war
   ```

3. Copy the war file from the binary volume on OIMHOST1 to any other hosts with a separate binaries volume.
   For example:

   ```
   cd /u01/oracle/products/12c/identity/idm/server/apps/
   scp oracle.iam.ui.custom-dev-starter-pack.war \
   iamoracle@OIMHOST2:/u01/oracle/products/14c/identity/idm/server/apps/.
   ```

Starting the Servers on OIMHOST2

After you upgrade Oracle Identity Manager on OIMHOST2, restart the servers.

Follow the same process that you used to start the servers on OIMHOST1. For instructions, see Performing OIM Bootstrap on OIMHOST1.

For information about stopping the servers and processes, see Stopping Servers and Processes.

Post-Upgrade Task

After performing the upgrade of Oracle Access Manager to 14c (14.1.2.1.0), you should complete the tasks summarized in this section, if required.

This section includes the following topics:

- Copying Custom Configurations
- Handling Custom Applications
- Reinstalling the ADF DI Excel Plug-in
  After you upgrade Oracle Identity Manager to 14c (14.1.2.1.0), uninstall and reinstall the ADF DI Excel plug-in, and then re-download the Excel.
- Defining System Properties for Legacy Connectors
- Increasing the Maximum Message Size for WebLogic Server Session Replication
- Increasing the maxdepth Value in setDomainEnv.sh
- Changing the JMS and TLOG Persistence Store After the Upgrade

Copying Custom Configurations

If you had set custom configuration in your 12c (12.2.1.4.0) Oracle home, you need to copy the custom configuration present in your backup of 12c (12.2.1.4.0) Oracle home to the 14c (14.1.2.1.0) Oracle home.

For example: Copy any contents from standard directories such as `XLIntegrations`, `connectorResources`, and so on, under the backup of 12c (12.2.1.4.0) Oracle home to the corresponding directories under the 14c (14.1.2.1.0) Oracle home.

Similarly, if your schedule job parameters are referring anything from the 12c (12.2.1.4.0) Oracle home, then copy them from the backup of 12c (12.2.1.4.0) Oracle home to the corresponding directories under the 14c (14.1.2.1.0) Oracle home.

> **Note:**
>
> The back up of custom configurations that you created in Backing up the 12c (12.2.1.4.0) Oracle Home Folder on OIMHOST are restored in this step.

Handling Custom Applications

If custom applications and libraries are present in your deployment of Oracle Identity Governance (OIG) 12c (12.2.1.4.0), Oracle recommends you to update them manually after the upgrade to Oracle Identity Governance (OIG) 14c (14.1.2.1.0).

Reinstalling the ADF DI Excel Plug-in

After you upgrade Oracle Identity Manager to 14c (14.1.2.1.0), uninstall and reinstall the ADF DI Excel plug-in, and then re-download the Excel.

Defining System Properties for Legacy Connectors

As part of post-upgrade tasks, for legacy connectors such as Resource Access Control Facility (RACF) that use the `tcITResourceInstanceOperationsBean.getITResourceInstanceParameters` method, you should create the following two system properties and update their values to `True`:

- `Service Account Encrypted Parameter Value`
- `Service Account Parameters Value Store`

For more information about these system properties, see Table 18-2 of section Non-Default System Properties in Oracle Identity Governance in *Administering Oracle Identity Governance*.

Oracle recommends creating these system properties only if a legacy connector or an old custom code requires the legacy behavior.

Increasing the Maximum Message Size for WebLogic Server Session Replication

Oracle recommends you to modify the Maximum Message Size from the default value of 10 MB to 100 MB. This value is used to replicate the session data across the nodes. You should perform this step for all the Managed servers and the Administration server.

1. Log in to the WebLogic Server Administration Console.
2. Navigate to **Servers**, select **Protocols**, and then click **General**.
3. Set the value of **Maximum Message Size** to `100 MB`.

Increasing the `maxdepth` Value in setDomainEnv.sh

The recommended value for the `maxdepth` parameter is `250`. To update this value:

1. Open the `$DOMAIN_HOME/bin/setDomainEnv.sh` file in a text editor.
2. Locate the following code block:

```
ALT_TYPES_DIR="${OIM_ORACLE_HOME}/server/loginmodule/wls,$
{OAM_ORACLE_HOME}/a
gent/modules/oracle.oam.wlsagent_11.1.1,${ALT_TYPES_DIR}"
export ALT_TYPES_DIR
CLASS_CACHE="true"
export CLASS_CACHE
```

3. Add the following lines at the end of the above code block:

```
JAVA_OPTIONS="${JAVA_OPTIONS} -Dweblogic.oif.serialFilter=maxdepth=250"
export JAVA_OPTIONS
```

4. Save and close the `setDomainEnv.sh` file.

Changing the JMS and TLOG Persistence Store After the Upgrade

The JMS and TLOG persistent store remain the same after the upgrade to Oracle Identity Manager 14c (14.1.2.1.0). That is, if the persistence store is file-based prior to the upgrade, it will be file-based after the upgrade as well.

If you want to change the persistence stores from a file-based system to a database-based system, you have to perform the steps manually. See Using Persistent Stores for TLOGs and JMS in an Enterprise Deployment.

Part II

Out-of-Place Upgrade of Oracle Identity Manager

In an out-of-place upgrade, you will create a new system and migrate the data from your existing system to the new system. You can perform an out-of-place upgrade from 12c (12.2.1.4.0) to Oracle Identity Manager 14c (14.1.2.1.0) environment by using the procedure described in this part.

This part contains the following topic:

- Performing an Out-of-Place Upgrade of Oracle Identity Manager
  The starting points for an out-of-place upgrade to Oracle Identity Manager 14c (14.1.2.1.0) is Oracle Identity Manager 12c (12.2.1.4.0).

5

Performing an Out-of-Place Upgrade of Oracle Identity Manager

The starting points for an out-of-place upgrade to Oracle Identity Manager 14c (14.1.2.1.0) is Oracle Identity Manager 12c (12.2.1.4.0).

To prepare for the upgrade of Oracle Identity Manager, verify that your system meets the basic requirements discussed in Pre-Upgrade Assessments.

This chapter includes the following topics:

- Pre-Upgrade Assessments
  Before starting the out-of-place upgrade of Oracle Identity Manager, you must check the cross-product interoperability and compatibility, system requirements, and certification requirements.
- Migrating Entities from 12c to 14c
  After you have installed the OIG 12c environment as per your requirements, migrate the following entities from 12c to 14c environment:
- Post Upgrade Steps
  As part of the post upgrade steps, you should follow the tuning guidelines and complete the sanity test.

Pre-Upgrade Assessments

Before starting the out-of-place upgrade of Oracle Identity Manager, you must check the cross-product interoperability and compatibility, system requirements, and certification requirements.

Install the 14c (14.1.2.1.0) version of Oracle Identity Governance as per your requirements (large, medium, or small deployment) on new hardware.

For installation instructions, see Installing and Configuring the Oracle Identity Governance Software. You must configure the new system by integrating components, as necessary.

The pre-upgrade check includes reviewing the current OIM environment (depending on the starting point) before starting the upgrade to OIM 14c (14.1.2.1.0), and then creating a list of features or components currently being used, such as OIM workflows, connectors, provisioning, targets, workflow policies, and admin roles/capabilities.

For more information, see Pre-Upgrade Requirements.

Migrating Entities from 12c to 14c

After you have installed the OIG 12c environment as per your requirements, migrate the following entities from 12c to 14c environment:

- Organizations
- Connectors
- Accounts

- Roles (Role, Role Membership, and Categories)
- User Records
- User Customizations
- Others

Organizations

Following options are available to migrate Organization records from the current OIM 11g environment (11.1.2.3 or 11.1.2.2) to 12c:

Option 1- Organization Bulk Load Utility

This option involves creating a source database table or a CSV file that contains the data you want to migrate.

For more information on using CSV files or creating database tables, see Creating the Input Source for the Bulk Load Operation in *Developing and Customizing Applications for Oracle Identity Governance*.

Option 2- Export And Import Feature In Sysadmin Console

After you have created your source data, you need to import the source data into the new 12c target system. For more information, see Migrating Incrementally Using the Deployment Manager.

Connectors

You should review the latest version of the connector available for 12c and use Application on Boarding (AoB) to create such connectors.

A new installation enables you to upgrade your targets to newer versions that are certified with 12c connectors.

If 12c connectors are not available, you can export or import existing user data as long as those connectors are supported in the 12c OIM server.

For more information, see Oracle Identity Governance 12c Connectors documentation.

For downloading connectors, see the Oracle Identity Governance Connector Downloads page.

For certification information for Oracle Identity Manager Connectors, see Oracle Identity Governance Connectors Certification.

> **Note:**
>
> If the connectors installed on 11g have no 12c version, you must check the certification, and then upgrade the existing connector to make it compatible with OIG 12c.

Accounts

After you set up the connectors as applications, you should start loading the account data from the target systems.

> **Note:**
>
> Target systems are applications such as database, LDAP, and so on, which OIM connects to using the OIM connectors.

Following options are available to load your accounts:

- **Option 1**: If the target system has account data, you can bulk load the account details (or data) by using the Bulk Load Utility. See Loading Account Data in *Developing and Customizing Applications for Oracle Identity Governance* guide.
- **Option 2**: You can load the target system account data into the new environment by using connector the reconciliation jobs.
- **Option 3**: You can use a flat file to load the data, similar to bulk load but using AoB directly. See Configuring Flat Files in *Performing Self Service Tasks with Oracle Identity Governance*.

Roles (Role, Role Membership, and Categories)

You can use the OIM Bulk Load Utility to import roles, role membership, and categories from a table or a CSV file. Export the relevant data files from the source OIM database.

For information on how to export and import this data, see Loading Role, Role Hierarchy, Role Membership, and Role Category Data in *Developing and Customizing Applications for Oracle Identity Governance*.

User Records

Following options are available to migrate user records from current OIM 11g (11.1.2.3 or 11.1.2.2) environment to 12c:

- **Option 1 - User Bulk Load Utility**

  This option includes exporting the user records to a table or a CSV file that will act as a source. See Loading OIM User Data in *Developing and Customizing Applications for Oracle Identity Governance* guide.
- **Option 2 - Trusted Recon of Users from 11g to 12c**

  This option includes using the Database User Management (DBUM) connector or a flat file connector to migrate the user records.
- **Option 3 - Data Load Using Flat Files**

  If the trusted source is an AoB application, this option includes loading data using flat files in AoB directly. See Configuring Flat Files in *Performing Self Service Tasks with Oracle Identity Governance*.

> **Note:**
>
> You cannot migrate user passwords by using the above options. You can set up SSO or LDAP as an authentication provider.

User Customizations

If you have added the custom User Defined Fields (UDF) in OIM 11g, you must create those UDFs in 12c as well.

> **WARNING:**
>
> Oracle does not support UDF migration (Deployment Manager and ADF Sandboxes).

> **Note:**
>
> To check if import or export from 11g to 12c works, export the user metadata from the 11g environment and import it to 12c, get the corresponding ADF sandbox, and then import it to 12c.

Others

You can also migrate the following items from your 11g environmen to the 12c environment by using the Export/Import option in the sysadmin console:

- Access policies
- Admin roles
- Application instances
- Approval policies
- Catalog UDFs
- Certification configurations
- Certification definitions
- Custom resource bundles
- E-mail definitions
- Error codes
- Event handlers
- Identity Audit configuration
- Identity Audit rules
- Identity Audit scan definitions
- IT resource definition
- IT resources
- JAR files
- Lookup definitions
- Notification templates
- Organization metadata

- Organizations
- Password policies
- Policies
- Plug-ins
- Prepopulation adapters
- Process definitions
- Process forms
- Provisioning workflows and process task adapters
- Request datasets
- Resource objects
- Risk configuration
- Role metadata
- Roles
- Scheduled jobs
- Scheduled tasks
- System properties
- User metadata

For more information, see Moving from a Test to a Production Environment and Using the Movement Scripts in the *Fusion Middleware Administrator's Guide*.

Post Upgrade Steps

As part of the post upgrade steps, you should follow the tuning guidelines and complete the sanity test.

- Tuning Considerations
- Performing a Sanity Test
- Reinstalling the ADF DI Excel Plug-in
  After you upgrade Oracle Identity Manager to 14c (14.1.2.1.0), uninstall and reinstall the ADF DI Excel plug-in, and then re-download the Excel.
- Defining System Properties for Legacy Connectors
- Increasing the Maximum Message Size for WebLogic Server Session Replication
- Increasing the maxdepth Value in setDomainEnv.sh

Tuning Considerations

Follow the performance tuning guidelines provided in the tuning documentation. See Oracle Identity Governance Performance Tuning.

Also, you should check the existing 11g system for custom indexes and create them in the 12c system.

Performing a Sanity Test

Perform a sanity test to ensure that the software and processes have been successfully upgraded and the system performs as expected. See Tab 5 of Doc ID 2667893.2.

Reinstalling the ADF DI Excel Plug-in

After you upgrade Oracle Identity Manager to 14c (14.1.2.1.0), uninstall and reinstall the ADF DI Excel plug-in, and then re-download the Excel.

Defining System Properties for Legacy Connectors

As part of post-upgrade tasks, for legacy connectors such as Resource Access Control Facility (RACF) that use the `tcITResourceInstanceOperationsBean.getITResourceInstanceParameters` method, you should create the following two system properties and update their values to `True`:

- `Service Account Encrypted Parameter Value`
- `Service Account Parameters Value Store`

For more information about these system properties, see Table 18-2 of section Non-Default System Properties in Oracle Identity Governance in *Administering Oracle Identity Governance*.

Oracle recommends creating these system properties only if a legacy connector or an old custom code requires the legacy behavior.

Increasing the Maximum Message Size for WebLogic Server Session Replication

Oracle recommends you to modify the Maximum Message Size from the default value of 10 MB to 100 MB. This value is used to replicate the session data across the nodes. You should perform this step for all the Managed servers and the Administration server.

1. Log in to the WebLogic Server Administration Console.
2. Navigate to **Servers**, select **Protocols**, and then click **General**.
3. Set the value of **Maximum Message Size** to `100 MB`.

Increasing the `maxdepth` Value in setDomainEnv.sh

The recommended value for the `maxdepth` parameter is `250`. To update this value:

1. Open the `$DOMAIN_HOME/bin/setDomainEnv.sh` file in a text editor.
2. Locate the following code block:

```
ALT_TYPES_DIR="${OIM_ORACLE_HOME}/server/loginmodule/wls,$
{OAM_ORACLE_HOME}/a
gent/modules/oracle.oam.wlsagent_11.1.1,${ALT_TYPES_DIR}"
export ALT_TYPES_DIR
CLASS_CACHE="true"
export CLASS_CACHE
```

3. Add the following lines at the end of the above code block:

```
JAVA_OPTIONS="${JAVA_OPTIONS} -Dweblogic.oif.serialFilter=maxdepth=250"
export JAVA_OPTIONS
```

4. Save and close the `setDomainEnv.sh` file.

Part III

Out-of-Place Cloned Upgrade of Oracle Identity Manager

In an out-of-place cloned upgrade, you will create a copy of your existing system on new hardware, and then perform an in-place upgrade on the clone. You can perform an out-of-place cloned upgrade of Oracle Identity Manager by using the procedure described in this part.

This part contains the following chapter:

- Performing an Out-of-Place Cloned Upgrade of Oracle Identity Manager
  The out-of-place upgrade procedure discussed in this guide explains how to perform a cloned upgrade of Oracle Identity Manager 12c (12.2.1.4.0) to Oracle Identity Manager 14c (14.1.2.1.0).

6

Performing an Out-of-Place Cloned Upgrade of Oracle Identity Manager

The out-of-place upgrade procedure discussed in this guide explains how to perform a cloned upgrade of Oracle Identity Manager 12c (12.2.1.4.0) to Oracle Identity Manager 14c (14.1.2.1.0).

This chapter includes the following topics:

- Pre-Upgrade Assessments
  The pre-upgrade check includes reviewing your current OIM 12c (12.2.1.4.0) environment before starting the cloned upgrade to OIM 14c (14.1.2.1.0).
- Performing an Out-of-Place Cloned Upgrade
  An out-of-place upgrade from Oracle Identity Manager 12c (12.2.1.4.0) to 14c (14.1.2.1.0) includes preparing the host files, cloning the database, binaries, and the configuration, and then upgrading the target system.
- Increasing the Maximum Message Size for WebLogic Server Session Replication
  As part of the post-upgrade tasks, Oracle recommends you to modify the Maximum Message Size from the default value of 10 MB to 100 MB. This value is used to replicate the session data across the nodes.
- Increasing the maxdepth Value in setDomainEnv.sh

Pre-Upgrade Assessments

The pre-upgrade check includes reviewing your current OIM 12c (12.2.1.4.0) environment before starting the cloned upgrade to OIM 14c (14.1.2.1.0).

For more information, see the following topics:

- Checking the Supported Versions
- Source Environment Validation for Use of Host Names
- Purging Unused Data
  Purging unused data and maintaining a purging methodology before an upgrade can optimize the upgrade process.

Checking the Supported Versions

You can upgrade the Oracle Identity Manager 12c (12.2.1.4.0) to 14c (14.1.2.1.0). You must make sure that OIM is fully patched with the latest bundle and required patches.

If you are running an older version of OIM, you must first upgrade it to 12c (12.2.1.4.0), and then to 14c (14.1.2.1.0).

Source Environment Validation for Use of Host Names

The cloning solution provided in this chapter relies on the use of host names and not IP addresses in all configuration properties. Validate the various domain and application

configuration parameters in the source environment to ensure that there are no IP addresses directly configured. If IP addresses are found to be in use, Oracle recommends you to update the source environment prior to beginning the cloning process.

This section includes the following topics:

- Auditing the WebLogic Server Domain Configuration
- Auditing the Application Configuration Data Stored in the Metadata Service (MDS)

Auditing the WebLogic Server Domain Configuration

Verify that the domain is not configured with IP addresses for the various listener, nodemanager, datasource host/SCAN/ONS parameters, and so on. As customer configurations vary in scope and the number of parameters are too many to enumerate specifically, only a basic audit process is provided here. A simple search of the domain configuration files for each known hostname, or by domain name, IP address list, or network range will provide a quick report.

The source environment might have host records such as:

```
# On-Prem Host Entries
10.99.5.42   srchost27.example.com srcHost27   webhost1
10.99.5.43   srchost28.example.com srcHost28   webhost2
10.99.5.44   srchost20.example.com srcHost20  ldaphost1
10.99.5.45   srchost21.example.com srcHost21  ldaphost2
10.99.5.46   srchost23.example.com srcHost23   oamhost1
10.99.5.47   srchost24.example.com srcHost24   oamhost2
10.99.5.48   srchost25.example.com srcHost25   oimhost1
10.99.5.49   srchost26.example.com srcHost26   oimhost2
# Compute VNIC Secondary IP for AdminServer floating VIPs
10.99.5.61 srcVIPiad.example.com srcVIPiad
10.99.5.62 srcVIPigd.example.com srcVIPigd
# Database Systems with on-prem override aliases
10.99.5.20 src-DB-SCAN.example.com src-DB-SCAN
# Load Balancer IP
10.99.5.6  prov.example.com  login.example.com  idstore.example.com
iadadmin.example.com  igdadmin.example.com  iadinternal.example.com
igdinternal.example.com
```

Values to check for can be written to a file for easy command-line use. Include the corporate network range, partial domain names, and partial strings from any corporate host naming convention that might be relevant, and then execute a search of all XML configuration files from the *DOMAIN_HOME*/config folder.

```
cat << EOF > /tmp/domainHostNameSearchList.txt
10.99.
.example.com
srcHost
webhohst
ldaphost
oamhost
oimhost
EOF
```

```
cd DOMAIN_HOME/config
find .-name "*.xml" -exec grep -H -f /tmp/domainHostNameSearchList.txt {} \;
```

This will result in a list of configuration *file paths/names*, and the line in which the text is found. The resulting list should include machine and listen-address entries, JDBC URLs, ONS Node list entries (if using Gridlink JDBC Drivers), and so on.

```
./config.xml:    <machine>OIMHOST1</machine>
./config.xml:    <listen-address>OIMHOST1</listen-address>
./config.xml:      <arguments>-Dtangosol.coherence.wka1=OIMHOST1 -
Dtangosol.coherence.wka2=OIMHOST2 -Dtangosol.coherence.localhost=OIMHOST1 -
Dtangosol.coherence.wka1.port=8089 -Dtangosol.coherence.wka2.port=8089 -
Dtangosol.coherence.localport=8089</arguments>
./config.xml:    <machine>OIMHOST1</machine>
./config.xml:    <listen-address>10.99.5.48</listen-address>
./config.xml:    <machine>OIMHOST1</machine>
./config.xml:    <listen-address>OIMHOST1</listen-address>
./config.xml:    <name>OIMHOST2</name>
./config.xml:      <name>OIMHOST2</name>
./config.xml:      <listen-address>srcHost26</listen-address>
./jdbc/mds-soa-jdbc.xml:
<url>jdbc:oracle:thin:@(DESCRIPTION=(ENABLE=BROKEN)
(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=src-DB-SCAN.example.com)
(PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=igdupgdb.example)))</url>
./jdbc/mds-soa-jdbc.xml:    <ons-node-list>src-DB-SCAN.example.com:6200</ons-
node-list>
```

Verify that all entries are using hostnames, either short or fully-qualified. These are the values that must be confirmed in the target host files.

> **Note:**
>
> Any configurations specifying IP addresses should be corrected in the source system prior to cloning.

Auditing the Application Configuration Data Stored in the Metadata Service (MDS)

Oracle Identity Governance stores configuration details in a Fusion Middleware Metadata Store (MDS) database schema. These configuration details include endpoint URI and JDBC connection strings that you should review and validate prior to cloning the environment. The hosts referenced in these URI and connection strings must be configured as hostnames or fully-qualified domain names (FQDN) rather than IP addresses. If IP addresses are used, they cannot be overridden in the target environment and you would have to change them during the cloning process.

Oracle recommends you to correct the source environment and replace any hard-coded IP addresses with appropriate host names prior to the cloning maintenance.

To audit the stored metadata configuration for OIM via WLST:

1. Log in to an OIM host in the source environment as the OS user with privileges to the *ORACLE_HOME* directory.

2. Create a temporary working directory.

```
mkdir -p /tmp/mds/oim/
```

3. Connect to the AdminServer via WLST.

```
$ ORACLE_HOME/common/bin/wlst.sh
wls:/offline> connect()
Please enter your username :weblogic
Please enter your password :
Please enter your server URL [t3://localhost:7001] :t3://ADMINHOST:7001
Connecting to t3://ADMINHOST:7001 with userid weblogic ...
Successfully connected to Admin Server 'AdminServer' that belongs to
domain 'IAMGovernanceDomain'.
wls:/IAMGovernanceDomain/serverConfig>
```

4. Export the OIM configuration XML data from the FMW Metadata Store and exit from WLST.

   - Application=*OIMMetadata*
   - server=*WLS_OIM1* (your server name may vary)
   - toLocation=`/tmp/mds/oim`
   - docs= `/db/oim-config.xml`

   For example:

```
wls:/IAMGovernanceDomain/serverConfig>
exportMetadata(application='OIMMetadata', server='WLS_OIM1',
toLocation='/tmp/mds/oim', docs='/db/oim-config.xml')

Executing operation: exportMetadata.

Operation "exportMetadata" completed. Summary of "exportMetadata"
operation is:
1 documents successfully transferred.
List of documents successfully transferred:

/db/oim-config.xml

wls:/IAMGovernanceDomain/serverConfig> exit()
```

5. Create a file of search terms to be used to filter for the relevant data from the OIM configuration. There are a lot of configuration elements in the exported XML file. Create a short list to use for filtering.

   For example:

```
$ cat << EOF > /tmp/mds/oim/grepHostValidationTerms.txt
<directDBConfigParams
bIPublisherURL
oimFrontEndURL
oimExternalFrontEndURL
oimJNDIURL
backOfficeURL
accessServerHost
```

```
tapEndpointUrl
soapurl
rmiurl
host
serviceURL
EOF
```

6. Search the OIM configuration data using the search terms.

   For example:

```
$ grep -f /tmp/mds_oim/grepHostValidationTerms.txt /tmp/mds/oim/db/oim-
config.xml

<directDBConfigParams checkoutTimeout="1200"
connectionFactoryClassName="oracle.jdbc.pool.OracleDataSource"
connectionPoolName="OIM_JDBC_UCP" driver="oracle.jdbc.OracleDriver"
idleTimeout="360" maxCheckout="1000" maxConnections="5" minConnections="2"
passwordKey="OIMSchemaPassword" sslEnabled="false"
url="jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=src-DB-
SCAN.example.com )(PORT=1521)) (CONNECT_DATA=
(SERVICE_NAME=igdupgdb.example)))" username="IGDUPG_OIM"
validateConnectionOnBorrow="true">
<bIPublisherURL>http://OIMHOST2:9704,OIMHOST1:9704</bIPublisherURL>
<oimFrontEndURL>http://igdinternal.example.com</oimFrontEndURL>
<oimExternalFrontEndURL>https://prov.example.com:443</
oimExternalFrontEndURL>
<oimJNDIURL>@oimJNDIURL</oimJNDIURL>
<backOfficeURL/>
<accessServerHost>srcHost23</accessServerHost>
<tapEndpointUrl>https://login.example.com:443/oam/server/dap/cred_submit</
tapEndpointUrl>
<soapurl>http://OIMHOST2:8001</soapurl>
<rmiurl>cluster:t3://cluster_soa</rmiurl>
<host>@oaacghost</host>
<serviceURL>@oaacgserviceurl</serviceURL>
```

7. Review the search results, verify all the configuration properties, and use appropriate hostnames or fully-qualified domain names.

> **Note:**
>
> - Some properties may have placeholder values (for example: *@oaacghost* or *@oaacgserviceurl*). These are acceptable.
> - The <`rmiurl`> URI specified is typically a WLS t3 protocol URI addressed to a WLS server name or cluster name, and does not use a hostname. This is also acceptable.

Purging Unused Data

Purging unused data and maintaining a purging methodology before an upgrade can optimize the upgrade process.

Some components have automated purge scripts. If you are using purge scripts, wait until the purge is complete before starting the upgrade process. The upgrade may fail if the purge scripts are running while using the Upgrade Assistant to upgrade your schemas.

Having excessive stale data in the database might cause problems when performing the upgrade schema updates. To optimize the upgrade process, it is recommended that you purge any stale or unnecessary data prior to the upgrade.

For instance, using data purge scripts included with OIM, as described in Using the Archival and Purge Utilities for Controlling Data Growth, allows your site to choose what data has to be archived into a different location, what data can be purged, and provides options to manage these operations.

> **Note:**
>
> In large systems with plenty of data, archiving/purging may take a long time. Oracle strongly recommends not to run the archival/purge scripts in parallel to improve performance.

Performing an Out-of-Place Cloned Upgrade

An out-of-place upgrade from Oracle Identity Manager 12c (12.2.1.4.0) to 14c (14.1.2.1.0) includes preparing the host files, cloning the database, binaries, and the configuration, and then upgrading the target system.

- Preparing the Host Files
- Cloning the Database
- Cloning the Oracle Binaries
- Cloning the Configuration
- Upgrading In-place Cloned Environment to 14c

Preparing the Host Files

In a cloned environment, the referenced host names in the target environment are the same as the host names in your source system. If you have followed the recommendations in the Enterprise Deployment Guide and used virtual host names for all configurations, this is simply a matter of aliasing these entries to the real target host names. For example:

```
10.0.2.17    oimhost1.idm.tenant.oraclevcn.com    oimhost1
```

If you are using physical host names in your source WebLogic configuration, you must alias these names to the real target host names. For example:

```
10.0.2.17   oimhost1.idm.tenant.oraclevcn.com   oimhost1
srchost25.example.com srcHost25
```

In addition, if the source environment has additional floating VIPs and FQDN for the AdminServer's Machine listen address and Node Manager host declaration, then the target Secondary IP addresses should be configured on the VNICs for the appropriate target compute instances and added to the hosts file. These secondary IP address entries should also include the source environment FQDNs and hostnames to override DNS when connecting to the AdminServer.

```
10.0.2.21 igdadminvhn.idm.tenant.oraclevcn.com igdadminvhn
srcVIPigd.example.com srcVIPigd
```

An example `/etc/hosts` file:

```
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6

# Compute with on-prem override aliases
10.0.2.11   webhost1.idm.tenant.oraclevcn.com   webhost1
srchost27.example.com srcHost27
10.0.2.12   webhost2.idm.tenant.oraclevcn.com   webhost2
srchost28.example.com srcHost28
10.0.2.13  ldaphost1.idm.tenant.oraclevcn.com  ldaphost1
srchost20.example.com srcHost20
10.0.2.14  ldaphost2.idm.tenant.oraclevcn.com  ldaphost2
srchost21.example.com srcHost21
10.0.2.15   oamhost1.idm.tenant.oraclevcn.com   oamhost1
srchost23.example.com srcHost23
10.0.2.16   oamhost2.idm.tenant.oraclevcn.com   oamhost2
srchost24.example.com srcHost24
10.0.2.17   oimhost1.idm.tenant.oraclevcn.com   oimhost1
srchost25.example.com srcHost25
10.0.2.18   oimhost2.idm.tenant.oraclevcn.com   oimhost2
srchost26.example.com srcHost26

# Compute VNIC Secondary IP for AdminServer floating VIPs
10.0.2.20 iadadminvhn.idm.tenant.oraclevcn.com iadadminvhn
srcVIPiad.example.com srcVIPiad
10.0.2.21 igdadminvhn.idm.tenant.oraclevcn.com igdadminvhn
srcVIPigd.example.com srcVIPigd

# Database Systems with on-prem override aliases
10.0.2.19  iamdbhost.idm.tenancy.oraclevcn.com  iamdbhost   src-DB-
SCAN.example.com src-DB-SCAN

# Load Balancer IP
10.0.1.10  prov.example.com  login.example.com  idstore.example.com
iadadmin.example.com  igdadmin.example.com  iadinternal.example.com
igdinternal.example.com
```

> **Note:**
>
> Ensure that the entries for each of the target compute instances and DB Host/SCAN addresses are present in the host file for all the hosts in the topology.

Cloning the Database

You can take a copy of your existing environment and then upgrade that copy. If you encounter issues during the upgrade, you will have the existing environment as a fallback.

For more information, see Performing an Upgrade via a Cloned Environment.

- Methods for Cloning Databases
- Cloning the Database Using the Export/Import Method
- Cloning the Database Using RMAN
- Cloning the Database Using Data Guard

Methods for Cloning Databases

There are different methods of cloning a database and each method has its own merits.

> **Note:**
>
> Oracle Identity and Access Management does not support Oracle Access Manager and Oracle Identity Manager configured to use the same database schema prefix. Before you upgrade, if both products co-exist and share the same database schemas, you must first split the database into two different prefixes and schema sets.

You can use the following options to clone the database:

Option 1 – Database Export Import

- Suitable for smaller sized databases.
- Allows movement between versions. Allows movement into Container Databases/Private Databases.
- Is a complete copy; redoing the exercise requires data to be deleted from the target each time.
- No ongoing synchronization.
- During cut-over the source system will need to be frozen for updates.

Option 2 – Duplicate Database Using RMAN

- Suitable for databases of any size.
- Takes a back up of an entire database.
- The database version and patch level should be the same on both the source and destination.
- Database upgrades will need to be performed as a separate task.

- CDP/PDB migration will have to be done as a separate exercise.
- No ongoing synchronization.
- During cut-over, you should freeze the source system for updates.

Option 3 – Data Guard Database

- Suitable for databases of any size.
- Takes a back up of an entire database.
- Database upgrades will need to be performed as a separate task.
- CDP/PDB migration will have to be done as a separate exercise.
- Ongoing synchronisation; Database can be opened to test the upgrade and closed again to keep data synchronized with the source system.

> **Note:**
>
> You should choose the solution based on your requirements.

Cloning the Database Using the Export/Import Method

On your 12c (12.2.1.4.0) environment, export the data from your database to an export file.

On the source environment:

1. Create and set the directory details for the export process on the source DB hosts.
   a. Make a directory on the source DB hosts in a location with sufficient space.

      ```
      mkdir -p /u01/installers/database
      ```

   b. On the source database, create a database directory object pointing to this location:

      ```
      SQL> CREATE DIRECTORY orcl_full AS '/u01/installers/database';
      ```

2. Shutdown WebLogic Server Managed Servers or Clusters for OIM, SOA, and BIP.

   > **Note:**
   >
   > If executing in parallel with the domain backup, coordinate the shut down of the entire domain including AdminServer and NodeManagers.

3. Stop the SOA DBMS queues in the source database.
   a. Connect as the SOAINFRA schema user and query for the user queues.

      ```
      $ sqlplus <PREFIX>_SOAINFRA@<sourceDB>
      SQL> COLUMN name FORMAT A32
      SQL> SELECT name,enqueue_enabled,dequeue_enabled
      FROM USER_QUEUES where queue_type = 'NORMAL_QUEUE' order by name;
      NAME                             ENQUEUE DEQUEUE
      -------------------------------- ------- -------
      B2B_BAM_QUEUE                     YES        YES
      ```

```
EDN_EVENT_QUEUE                   YES        YES
EDN_OAOO_QUEUE                    YES        YES
IP_IN_QUEUE                       YES        YES
IP_OUT_QUEUE                      YES        YES
TASK_NOTIFICATION_Q               YES        YES

6 rows selected.
```

b. Stop each queue.

```
SQL> BEGIN

DBMS_AQADM.STOP_QUEUE ('B2B_BAM_QUEUE');

DBMS_AQADM.STOP_QUEUE ('EDN_OAOO_QUEUE');

DBMS_AQADM.STOP_QUEUE ('EDN_EVENT_QUEUE');

DBMS_AQADM.STOP_QUEUE ('IP_IN_QUEUE');

DBMS_AQADM.STOP_QUEUE ('IP_OUT_QUEUE');

DBMS_AQADM.STOP_QUEUE ('TASK_NOTIFICATION_Q');

END;

/
exit
```

4. As the OIM schema user, query for and stop any running `DBMS_SCHEDULER` jobs in the source database.

```
$ sqlplus <PREFIX>_OIM@<sourceDB>

SQL> SELECT job_name,session_id,running_instance,elapsed_time
FROM user_scheduler_running_jobs ORDER BY job_name;

no rows selected
```

> **Note:**
>
> In case of any running jobs, either wait till the job is complete or stop the job ‘gracefully’ using:
>
> ```
> SQL> BEGIN
>
> DBMS_SCHEDULER.stop_job('REBUILD_OPTIMIZE_CAT_TAGS');
>
> END;
>
> /
> SQL> exit
> ```

5. Grant system policies to avoid errors during export datapump jobs.

```
$ sqlplus SYS as SYSDBA
SQL> GRANT EXEMPT ACCESS POLICY TO SYSTEM;
SQL> exit
```

6. Export the system and application schemas from the source database, setting the directory property appropriately.

   a. Export the `system.schema_version_registry` table and view:

```
$ expdp \"sys/<password>@<sourcedb> as sysdba \" \
     DIRECTORY=orcl_full \
     DUMPFILE=oim_system.dmp \
     LOGFILE=oim_system_exp.log \
     SCHEMAS=SYSTEM \
     INCLUDE= VIEW:"IN('SCHEMA_VERSION_REGISTRY')"
TABLE:"IN('SCHEMA_VERSION_REGISTRY$')"\
     JOB_NAME=MigrationExportSys
```

   b. Export all of the schemas used by the datasources in the source WebLogicServer domain.

```
$ expdp \"sys/<password>@<sourcedb> as sysdba \" \
     DIRECTORY=orcl_full \
     DUMPFILE=oim.dmp \
     LOGFILE=oim_exp.log \

SCHEMAS=<PREFIX>_OIM,<PREFIX>_SOAINFRA,<PREFIX>_BIPLATFORM,<PREFIX>_MDS,
<PREFIX>_ORASDPM,<PREFIX>_OPSS,IGDJMS,IGDTLOGS \
     JOB_NAME=MigrationExport \
     EXCLUDE=STATISTICS
```

7. Extract the source database DDL for the tablespaces, schema users, and grants.

   This step allows the efficient creation of the correct tablespaces on the target database and retains the schema user passwords. Therefore, domain reconfiguration is not necessary. System and Object grants for objects outside the exported schemas are also accounted for to reduce the risk of invalid objects and recompilation difficulties.

   An example script is provided to create the complete SQL DDL output all at once. The example will need to be modified if not using a CDB/PDB.

   a. In SQLPLUS, execute the example SQL script to extract the DDL to a `ddl.sql` file in the same directory as the datapump exported dumps. Enter the source environment and the target PDB. Output will be copied to both the screen and in the file named `ddl.sql`.

```
$ cd /u01/installers/database
$ sqlplus SYS as SYSDBA
SQL> @extract_ddl.sql
Enter RCU Prefix: <PREFIX>
Enter PDB: targetPDB
```

   Example SQL Script:

> **Note:**
>
> Lines in bold are applicable only if your target database is a PDB. This SQL assumes that all the objects are created using the RCU prefix. If you have created objects without the prefix (for example tablespaces/users for JMS or TLogs, add these manually).

```
$ cat << EOF > extract_ddl.sql
set pages 0
set feedback off
set heading off
set long 5000
set longchunksize 5000
set lines 200
set verify off
exec dbms_metadata.set_transform_param
(dbms_metadata.session_transform, 'SQLTERMINATOR', true);
exec dbms_metadata.set_transform_param
(dbms_metadata.session_transform, 'PRETTY', true);
accept PREFIX char prompt 'Enter RCU Prefix:'
accept PDBNAME char prompt 'Enter PDB:'

spool ddl.sql

select 'alter session set container=&&PDBNAME;'
from dual
/
SELECT DBMS_METADATA.GET_DDL('TABLESPACE',Tablespace_name)
from  dba_tablespaces
where tablespace_name like '&&PREFIX%'
/
set lines 600
SELECT DBMS_METADATA.GET_DDL('USER',USERNAME)
from DBA_USERS
where USERNAME like '&&PREFIX%'
/
set lines 200
SELECT DBMS_METADATA.GET_GRANTED_DDL ('SYSTEM_GRANT',USERNAME)
from DBA_USERS
where USERNAME like '&&PREFIX%'
and USERNAME NOT LIKE '%_IAU_APPEND'
and USERNAME NOT LIKE '%_IAU_VIEWER'
/

SELECT DBMS_METADATA.GET_GRANTED_DDL ('OBJECT_GRANT',USERNAME)
from DBA_USERS
where USERNAME like '&&PREFIX%'
and USERNAME NOT LIKE '%TLOGS'
and USERNAME NOT LIKE '%JMS'
/

spool off
EOF
```

b. Delete any object grants for system `QT*_BUFFER` views in the output `ddl.sql`.The buffer views will not exist in the target database and cause errors.

```
$ sed -i.bak -e '/QT.*_BUFFER/d' /u01/installers/database/ddl.sql
```

8. Re-start the SOA DBMS queues. Connect as the SOAINFRA schema user and restart each queue that was stopped earlier.

```
$ sqlplus <PREFIX>_SOAINFRA@sourceDB
SQL> BEGIN

DBMS_AQADM.START_QUEUE ('B2B_BAM_QUEUE');

DBMS_AQADM.START_QUEUE ('EDN_OAOO_QUEUE');

DBMS_AQADM.START_QUEUE ('EDN_EVENT_QUEUE');

DBMS_AQADM.START_QUEUE ('IP_IN_QUEUE');

DBMS_AQADM.START_QUEUE ('IP_OUT_QUEUE');

DBMS_AQADM.START_QUEUE ('TASK_NOTIFICATION_Q');

END;

/
SQL> COLUMN name FORMAT A32
SQL> SELECT name,enqueue_enabled,dequeue_enabled
FROM USER_QUEUES where queue_type = 'NORMAL_QUEUE' order by name;

NAME                             ENQUEUE DEQUEUE
-------------------------------- ------- -------
B2B_BAM_QUEUE                     YES        YES
EDN_EVENT_QUEUE                   YES        YES
EDN_OAOO_QUEUE                    YES        YES
IP_IN_QUEUE                       YES        YES
IP_OUT_QUEUE                      YES        YES
TASK_NOTIFICATION_Q               YES        YES

6 rows selected.
SQL> exit
```

9. Re-start the WebLogic Server Managed Servers or clusters for OIM, SOA, and BIP.
10. Replicate the DDL SQL and the datapump dump files to the target database host.
    - `oim.dmp`
    - `oim_system.dmp`
    - `ddl.sql`

On the target environment:

1. Install/configure the target database sufficiently in accordance with FMW requirements. Install a version of the Oracle database you want to use on the target environment. This database can be a single instance database, a real applications cluster (RAC) database, a

standard database, or a Container Database with OIG in a separate pluggable database (PDB).

2. Validate that the target database is configured to meet all the criteria of Oracle Identity Manager as defined in Installing and Configuring the Oracle Identity Governance Software in the *Installing and Configuring Oracle Identity and Access Management*.
3. Create the TNS entry for the Pluggable Database in the target system, if necessary. For example:

```
IGDPDB =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCP)
                 (HOST = iamdbhost.idm.tenancy.oraclevcn.com)
                 (PORT = 1521)
      )
      (CONNECT_DATA =
          (SERVER = DEDICATED)
          (SERVICE_NAME = igdpdb.idm.tenancy.oraclevcn.com)
      )
     )
```

4. Create and set the directory details for the export process on the source DB hosts.
   a. Make a directory on the target DB hosts in a location with sufficient space.

   ```
   $ mkdir -p /u01/installers/database
   ```

   b. Create a database directory object pointing to this location on the source and destination databases.

   ```
   SQL> CREATE DIRECTORY orcl_full AS '/u01/installers/database';
   ```

5. Create a database restore point in case there is a need to roll back the transaction.
6. Create and start a database service for the new database with the same service name as the source environment.
   For example:

```
$ srvctl add service -db iamcdb -pdb igdpdb -service onpremservice -
rlbgoal SERVICE_TIME -clbgoal SHORT
$ srvctl start service -db iamcdb -service onpremservice
$ srvctl status service -db iamcdb -service onpremservice
```

7. Confirm that the exported datapump dump files and SQL files are available on the target database host in the correct directory, and the DBA directory name and path in the database match.

```
$ ls -al /u01/installers/database
$ sqlplus / as sysdba
SQL> ALTER SESSION SET CONTAINER = igdpdb;
SQL> CREATE DIRECTORY orcl_full AS '/u01/installers/database';
```

   To verify:

```
$ sqlplus / as sysdba
SQL> ALTER SESSION SET CONTAINER = igdpdb;
```

```
SQL> COLUMN directory_name FORMAT A32
SQL> COLUMN directory_path FORMAT A64
SQL> set linesize  128
SQL> SELECT directory_name,directory_path FROM dba_directories ORDER BY
directory_name;
```

8. Confirm that the required `DBMS_SHARED_POOL` and `XATRANS` database objects exist and create them if they do not. Check for a count of '2' for each of the following SQLs on the target database where the OIM schema export dump is to be restored.

```
SQL> SELECT COUNT(*) FROM dba_objects
WHERE owner = 'SYS' AND object_name = 'DBMS_SHARED_POOL'
AND object_type IN ('PACKAGE','PACKAGE BODY');

  COUNT(*)
----------
         2

SQL> SELECT COUNT(*) FROM dba_objects
WHERE owner = 'SYS' AND object_name like '%XATRANS%';

  COUNT(*)
----------
         0
```

   a. If `DBMS_SHARED_POOL` count is < 2, run the appropriate SQL to re-configure:

```
SQL> @/u01/app/oracle/product/19.0.0.0/dbhome_1/rdbms/admin/dbmspool.sql
SQL> @/u01/app/oracle/product/19.0.0.0/dbhome_1/rdbms/admin/prvtpool.plb
```

   b. If `XATRANS` count is < 2, run the appropriate SQL to reconfigure:

```
SQL> @/u01/app/oracle/product/19.0.0.0/dbhome_1/rdbms/admin/xaview.sql
```

9. Import the source database system dump from the correct folder to create the `schema_version_registry` table and view, then create the required public synonym manually via SQL.

```
$ cd /u01/installers/database
$ impdp \"SYS/<password>@<targetdb> AS SYSDBA\" \
    PARALLEL=4
    DIRECTORY=orcl_full \
    DUMPFILE=oim_system.dmp \
    LOGFILE=oim_system_imp.log \
    FULL=YES;

$ sqlplus / as sysdba

SQL> alter session set container=igdpdb;
SQL> CREATE PUBLIC SYNONYM schema_version_registry FOR
system.schema_version_registry;
SQL> exit
```

10. Verify that the `schema_version_registry` table data matches your source environment. It is important to check that the following query returns rows that are consistent with your deployment. This table should have been imported as part of the above steps. If it fails to do so you must populate the table with values from your source system.

```
$ sqlplus / as sysdba
SQL> alter session set container=igdpdb;SQL> set linesize 100
SQL> col comp_id for a10
SQL> col comp_name for a50
SQL> col version for a10
SQL> select comp_id, comp_name, version, status, upgraded
from system.schema_version_registry;

Output will look something like:

COMP_ID    COMP_NAME                                          VERSION
STATUS      U
---------- -------------------------------------------------- ----------
---------- -
BIPLATFORM OracleBI and EPM                                   11.1.1.9.0
VALID       N
MDS        Metadata Services                                  11.1.1.9.0
VALID       N
OIM        Oracle Identity Manager                            11.1.2.3.0
VALID       N
OPSS       Oracle Platform Security Services                  11.1.1.9.0
VALID       N
ORASDPM    SDP Messaging                                      11.1.1.9.0
VALID       N
SOAINFRA   SOA Infrastructure Services                        11.1.1.9.0
VALID       N
```

11. Execute the DDL SQL from the source database to create the required tablespaces, schema users with the same passwords, system grants, and object grants. If using a PDB, ensure that you set the container correctly.

```
$ sqlplus / as sysdba
SQL> alter session set container=igdpdb;
SQL> @'/u01/installers/database/ddl.sql'
SQL> exit
```

12. Import the application schemas.

> **Note:**
>
> There will be ORA-31684 errors due to pre-created the users. Ignore the following types of errors:
>
> - Procedure/Package/Function/Trigger compilation warnings
> - DBMS_AQ errors
> - ORA-31684: Object type USER:"" already exists

For example:

```
$ cd /u01/installers/database
$ impdp \"SYS/<password>@<targetdb> AS SYSDBA\" \
    PARALLEL=4 \
    DIRECTORY=orcl_full \
    DUMPFILE=oim.dmp \
    LOGFILE=oim_imp.log
    FULL=YES;
```

13. Query for any invalid objects for the imported schemas and execute a recompile for each schema with invalid objects.
For example:

```
$ sqlplus / as sysdba
SQL> alter session set container=igdpdb;
SQL> COLUMN owner       FORMAT A24
SQL> COLUMN object_type FORMAT A12
SQL> COLUMN object_name FORMAT A32
SQL> SET LINESIZE 128
SQL> SET PAGESIZE 50

SQL> SELECT owner,object_type,object_name, status
FROM   dba_objects
WHERE  status = 'INVALID'
AND owner like '<PREFIX>'
ORDER BY owner, object_type, object_name;

OWNER                    OBJECT_TYPE  OBJECT_NAME
STATUS
------------------------ ------------ --------------------------------
-------
IGDUPG_OIM               SYNONYM      ALTERNATE_ADF_LOOKUPS
INVALID
IGDUPG_OIM               SYNONYM      ALTERNATE_ADF_LOOKUP_TYPES
INVALID
IGDUPG_OIM               SYNONYM      FND_LOOKUPS
INVALID
IGDUPG_OIM               SYNONYM      FND_STANDARD_LOOKUP_TYPES
INVALID

SQL> EXECUTE UTL_RECOMP.RECOMP_SERIAL('IGDUPG_OIM');

SQL> SELECT owner,object_type,object_name, status
FROM   dba_objects
WHERE  status = 'INVALID'
AND owner like '<PREFIX>'
ORDER BY owner, object_type, object_name;

no rows selected
```

14. Start the SOA DBMS queues.

a. Connect as the SOAINFRA schema user and query for the user queues.

```
$ sqlplus <PREFIX>_SOAINFRA@<sourceDB>
SQL> COLUMN name FORMAT A32
SQL> SELECT name,enqueue_enabled,dequeue_enabled  FROM USER_QUEUES
where queue_type = 'NORMAL_QUEUE' order by name;

NAME                             ENQUEUE DEQUEUE
-------------------------------- ------- -------
B2B_BAM_QUEUE                     YES        YES
EDN_EVENT_QUEUE                   YES        YES
EDN_OAOO_QUEUE                    YES        YES
IP_IN_QUEUE                       YES        YES
IP_OUT_QUEUE                      YES        YES
TASK_NOTIFICATION_Q               YES        YES

6 rows selected.
```

b. Start each queue.

```
SQL> BEGIN

DBMS_AQADM.START_QUEUE ('B2B_BAM_QUEUE');

DBMS_AQADM.START_QUEUE ('EDN_OAOO_QUEUE');

DBMS_AQADM.START_QUEUE ('EDN_EVENT_QUEUE');

DBMS_AQADM.START_QUEUE ('IP_IN_QUEUE');

DBMS_AQADM.START_QUEUE ('IP_OUT_QUEUE');

DBMS_AQADM.START_QUEUE ('TASK_NOTIFICATION_Q');

END;

/
exit
```

Cloning the Database Using RMAN

Clone the database from the source environment to the target environment by using RMAN. See Transferring Data with RMAN.

Cloning the Database Using Data Guard

You can manually create a physical standby database using Data Guard. See Creating a Physical Standby Database in *Oracle Data Guard Concepts and Administration*.

Cloning the Oracle Binaries

Use your preferred backup/restore tools to archive and transfer the MW_HOME binaries and OraInventory directories.

This section includes the following topic:

Using Backup/Restore Tools to Clone the Binaries

> **Note:**
>
> For this exercise, you can use any backup and restore tool you are familiar with. The example below uses the tar tool. But any command that can back up and restore directories and sub-directories can be used. You can take a back up with the domain and NodeManagers online or offline. However, Oracle recommends to execute the backup with all FMW processes shut down.

Take a backup:

Complete the following steps to take a backup of your source environment domain:

1. Using your preferred backup tool, take a backup of the following directories in the source environment:
   - `oraInventory`
   - `MW_HOME`

   For example, a command on `OAMHOST1` may appear as follows:

   ```
   tar cfzP /u01/oracle/backups/oamhost1_binaries.tar.gz /u01/oracle/
   oraInventory MW_HOME
   ```

2. Repeat the command on any supplementary nodes using the separate product binary volumes.

   > **Note:**
   >
   > When using the shared filesystem volumes for the Oracle products `MW_HOME` locations, you should take the binary backups from only one host per volume.

   For example, a command on `OAMHOST2` may appear as follows:

   ```
   tar cfzP /u01/oracle/backups/oamhost2_binaries.tar.gz /u01/oracle/
   oraInventory MW_HOME
   ```

3. Copy the resulting backup files to their appropriate target environment hosts.

Restore the backup

Using your preferred extraction tool, extract the backup to your target environment nodes.

> **Note:**
>
> When using the shared filesystem volumes for the Oracle products `MW_HOME` locations, you should restore the binary backups to only one host per volume.

For example:

On OAMHOST1, run the following command:

```
tar xvfzP oamhost1.tar.gz
```

On OAMHOST2, run the following command:

```
tar xvfzP oamhost2.tar.gz
```

Cloning the Configuration

Use your preferred backup/restore tools to clone the configuration.

This section includes the following topics:

- Using Backup/Restore Tools to Clone the Oracle Identity Manager Domain
- Starting the OIM Domain
- Executing the OIM LDAP Consolidated Full Reconciliation Job

Using Backup/Restore Tools to Clone the Oracle Identity Manager Domain

> **Note:**
>
> For this exercise, you can use any backup and restore tool you are familiar with. The example below uses the tar tool. But any command that can back up and restore directories and sub-directories can be used. You can take a back up with the domain and NodeManagers online or offline. However, Oracle recommends to execute the backup with all FMW processes shutdown.

Take a backup:

Perform the following steps to take a backup of the source environment binaries and Oracle Inventory:

1. Using your preferred backup tool, take a backup of the following locations from OIMHOST1 on the source site:
   - `oraInventory`
   - `Nodemanager`
   - Application Server domain home (`ASERVER_HOME`)
   - Managed Server domain home if you have a separate location as described in the Enterprise Deployment Guide (`MSERVER_HOME`)
   - `Keystores`
   - Runtime directories

> **Note:**
>
> If you have a combined DOMAIN_HOME rather than a segregated one, as described in the Enterprise Deployment Guide, include DOMAIN_HOME rather than ASERVER_HOME and MSERVER_HOME.

For example, a command on OIMHOST1 may appear as follows:

```
tar cvzPpsf oimhost1.tar.gz \
   /u01/oracle/oraInventory \
   /u01/oracle/config/nodemanager/OIMHOST1 \
   /u01/oracle/config/nodemanager/OIMHOST2 \
   /u01/oracle/config/nodemanager/IGDADMINVHN \
   /u01/oracle/config/keystores \
   /u01/oracle/runtime/domains/IAMGovernanceDomain \
   /u01/oracle/config/domains/IAMGovernanceDomain \
   /u02/private/oracle/config/domains/IAMGovernanceDomain
```

2. Repeat the command on any supplementary nodes. For example, a command on OIMHOST2 may appear as follows:

```
tar cvzPpsf OIMHOST2.tar.gz /u02/private/oracle/config/domains/
IAMGovernanceDomain
```

3. Copy the resulting backup files to their appropriate target environment hosts.
4. Delete any lock and log files in the domain that have been replicated from the source environment.
   - Remove any lock files for all NodeManager folders on the appropriate cloned environment hosts by running the following command:

     ```
     find /u01/oracle/config/nodemanager -type f -name "*.lck" -exec rm -f {}
     \;
     ```

   - Remove any lock files from the ASERVER_HOME and MSERVER_HOME folders on the appropriate cloned environment hosts by running the following command:

     > **Note:**
     >
     > If you have a combined DOMAIN_HOME rather than a segregated one as described in the Enterprise Deployment Guide, include DOMAIN_HOME rather than ASERVER_HOME and MSERVER_HOME.

     For example, on OIMHOST1, run the following command:

     ```
     # Lock Files Cleanup:

     find /u01/oracle/config/nodemanager -type f -name "*.lck" -exec rm -f
     {} \;

     find  /u01/oracle/config/domains/IAMGovernanceDomain \
         -type f  \( -name "*.lck" -or -name "*.lok" \) -print -exec rm -f
     ```

```
{} \;

find  /u02/private/oracle/config/domains/IAMGovernanceDomain \
    -type f  \( -name "*.lck" -or -name "*.lok" \) -print -exec rm -f
{} \;

# Log File Cleanup:

find /u01/oracle/config/nodemanager/OIMHOST1 \
    -type f \( -name '*.log' -or -name '*.out' \) -print -exec rm -f {}
\;

find /u01/oracle/config/nodemanager/OIMHOST2 \
    -type f \( -name '*.log' -or -name '*.out' \) -print -exec rm -f {}
\;

find /u01/oracle/config/nodemanager/IGDADMINVHN \
    -type f \( -name '*.log' -or -name '*.out' \) -print -exec rm -f {}
\;

find ${ASERVER_HOME}/servers/AdminServer/logs \
    -type f ! -size 0c -print -exec rm -f {} \+

find ${MSERVER_HOME}/servers/*/logs \
    -type f ! -size 0c -print -exec rm -f {} \+
```

For example, on `OIMHOST2`, run the following command:

```
# Lock Files Cleanup:

find  /u02/private/oracle/config/domains/IAMGovernanceDomain \
    -type f  \( -name "*.lck" -or -name "*.lok" \) -print -exec rm -f
{} \;

# Log File Cleanup:

find ${MSERVER_HOME}/servers/*/logs \
    -type f ! -size 0c -print -exec rm -f {} \+
```

- Optionally, remove the old log files from the `NodeManager` and Managed Server folders in the cloned domain:

  For example, on `OIMHOST1`, run the following command:

```
find /u01/oracle/config/nodemanager/OIMHOST1 \
    -type f \( -name '*.log' -or -name '*.out' \) -print -exec rm -f {}
\;
find /u01/oracle/config/nodemanager/OIMHOST2 \
    -type f \( -name '*.log' -or -name '*.out' \) -print -exec rm -f {}
\;

find /u01/oracle/config/nodemanager/IGDADMINVHN \
    -type f \( -name '*.log' -or -name '*.out' \) -print -exec rm -f {}
\;

find ASERVER_HOME/servers/AdminServer/logs \
```

```
    -type f ! -size 0c -print -exec rm -f {} \+

find MSERVER_HOME/servers/*/logs \
    -type f ! -size 0c -print -exec rm -f {} \+
```

For example, on `OIMHOST2`, run the following command:

```
find MSERVER_HOME/servers/*/logs \ -type f ! -size 0c -print -exec rm -f
{} \+
```

Restore the backup in the cloned environment

Using your preferred extraction tool, extract the backup to your target environment nodes.

> **Note:**
>
> If using tar, be sure to preserve permissions and root paths.

For example:

On OIMHOST1, run the following command:

```
tar xvzPpsf oimhost1.tar.gz
```

On OIMHOST2, run the following command:

```
tar xvzPpsf oimhost2.tar.gz
```

Starting the OIM Domain

After successfully restoring the backup to the target environment instances, do the following to start the domain:

- Start the Node Manager for the *ASERVER_HOME*.
- Start the Node Manager for the *MSERVER_HOME* on all nodes.

> **Note:**
>
> If you have a single *DOMAIN_HOME*, start the Node Manager associated with that *DOMAIN_HOME*.

- Start the Administration Server and check logs.
- Start the SOA Managed Server/Cluster and check logs.
- Start the Business Intelligence Platform Managed Server/Cluster and check logs.
- Start the OIM Managed Server/Cluster and check logs.

Executing the OIM LDAP Consolidated Full Reconciliation Job

After cloning the domain, a full reconciliation job needs to be executed. See Jobs in *Administrator's Guide for Oracle Identity Manager*.

To execute the reconciliation job:

> **Note:**
>
> You have to perform the reconciliation job only if the 12c (12.2.1.4.0) setup is using LDAP Connectors. This step is not required if the setup is using LDAPSync because LDAPSync will be disabled after the upgrade is complete.

1. Log in to `https://igdadmin.example.com/sysadmin` and authenticate as *xelsysadm*.
2. In the left-pane, under **System Configuration**, click **Scheduler**. A popup window will appear.
3. In the Identity System Administration popup window, search for the scheduled job: *LDAP Consolidated Full Reconciliation*.
4. Click the *LDAP Consolidated Full Reconciliation* entry in the search results to view the job details.
5. Click **Run Now** to execute the job and verify the confirmation message: `Job is running`.
6. Periodically click the **Refresh** button and verify the job status.
7. When the Job Status shows `Stopped`, validate the Execution Status for `Success`. Check logs and troubleshoot as needed.
8. Click the **Event Management** tab and execute an empty search for all recent reconciliation events.
9. Spot-check the events to assure that the current status is either `Creation Succeeded` or `Update Succeeded`.

Upgrading In-place Cloned Environment to 14c

After cloning the 12c (12.2.1.4.0) domain to the target system, you can upgrade the target system to Oracle 14c (14.1.2.1.0). For instructions, see:

- For highly available environments, see Upgrading Oracle Identity Manager Highly Available Environments.
- For single node environments, see Upgrading Oracle Identity Manager Single Node Environments.

Increasing the Maximum Message Size for WebLogic Server Session Replication

As part of the post-upgrade tasks, Oracle recommends you to modify the Maximum Message Size from the default value of 10 MB to 100 MB. This value is used to replicate the session data across the nodes.

You should perform this step for all the Managed servers and the Administration server.

1. Log in to the WebLogic Server Administration Console.
2. Navigate to **Servers**, select **Protocols**, and then click **General**.
3. Set the value of **Maximum Message Size** to `100 MB`.

Increasing the `maxdepth` Value in setDomainEnv.sh

The recommended value for the `maxdepth` parameter is `250`. To update this value:

1. Open the `$DOMAIN_HOME/bin/setDomainEnv.sh` file in a text editor.
2. Locate the following code block:

```
ALT_TYPES_DIR="${OIM_ORACLE_HOME}/server/loginmodule/wls,$
{OAM_ORACLE_HOME}/a
gent/modules/oracle.oam.wlsagent_11.1.1,${ALT_TYPES_DIR}"
export ALT_TYPES_DIR
CLASS_CACHE="true"
export CLASS_CACHE
```

3. Add the following lines at the end of the above code block:

```
JAVA_OPTIONS="${JAVA_OPTIONS} -Dweblogic.oif.serialFilter=maxdepth=250"
export JAVA_OPTIONS
```

4. Save and close the `setDomainEnv.sh` file.

A

Troubleshooting the Oracle Identity Manager Upgrade

If you encounter errors during or after the upgrade of Oracle Identity Manager to 14c (14.1.2.1.0), review the following troubleshooting procedures.

- Error CFGFWK-60953: Application or library was not relocated to the new MW home
  An error in the reconfiguration templates can result if there are deployments remaining in the original Middleware home. You must delete the deployments from the Middleware home before running the Reconfiguration Wizard.
- Reconfig.sh OPSS Processing Phase Generates an ORA-00001
  The OPSS schema uses sequences to generate next values for some of its tables. If one or some of the sequences next values are lower than the value maximum value in the tables, then they need to be changed.
- Oracle Identity Manager Server Throws OutOfMemoryError
  When you start the servers post upgrade, `OutOfMemoryError` is thrown.
- Failure in UPDATE_WORKFLOW_POLICIES Post-Bootstrap Task
  The `UPDATE_WORKFLOW_POLICIES` post-bootstrap task fails when you start the OIM Managed server after the upgrade.
- MDS Customizations are Removed After You Restart the OIM Managed Server of an Upgraded Setup
  If any MDS customizations are done after a successful upgrade to 14c (14.1.2.1.0) and if those customizations are lost after you restart the OIM Managed Server, you cannot recover the MDS changes. You have to do the MDS customizations again.
- OPatch Fails for not Finding the 'fuser' Command
  OPatch fails when it is unable to locate the `fuser` command.
- Administration Server Has a Slow Start After the Upgrade
  The Administration Server experiences a slow start after the upgrade.
- NPE Encountered on Starting OIM Server After Running the Upgrade Assistant
  A Null Pointer Exception (NPE) is encountered when starting the OIM server after running the Upgrade Assistant for upgrading the domain configuration.
- OIM Bootstrap Fails Due to the Presence of Custom Application JARs
  If there are any custom developed libraries or JARs placed inside the *OIM_HOME*, the OIM bootstrap fails during the upgrade to Oracle Identity Manager 14c (14.1.2.1.0).
- Incorrect Links in Password Reset Emails
  The OIG system generated password reset email has links in the `applewebdata://<ANY_RANDOM_GUID>/null` format, which is incorrect.

Error CFGFWK-60953: Application or library was not relocated to the new MW home

An error in the reconfiguration templates can result if there are deployments remaining in the original Middleware home. You must delete the deployments from the Middleware home before running the Reconfiguration Wizard.

While executing the Reconfiguration Wizard on a domain that was created in 12.2.1.3.0 and then upgraded to 12.2.1.4.0, errors can occur if there are extraneous deployments remaining in the domain.

Correct the reconfiguration template as detailed in the error. In this case it is the ""jax-rs(2,2.22.4.0)" library and this is only seen when the environment being upgraded to 14c was upgraded from 12.2.1.3.0 to 12.2.1.4.0.

1. Using the 12c WebLogic Admin Console, access and log into the WLS admin Console for the OAM WebLogic Domain
2. Select "Deployments"
3. Navigate the deployment called "jax-rs(2,2.22.4.0)" and select the checkbox.
4. Click **Delete**.

Reconfig.sh OPSS Processing Phase Generates an ORA-00001

The OPSS schema uses sequences to generate next values for some of its tables. If one or some of the sequences next values are lower than the value maximum value in the tables, then they need to be changed.

In the OPSS schema run the following:

```
SELECT sequence_name, last_number FROM all_sequences WHERE sequence_owner =
'<Prefix>_OPSS';
```

The `SELECT sequence_name.....` query will show what the database sees as the next sequence number to be used for the various tables.

Run the following queries:

```
Select max(entryid) from jps_dn;
Select max(jps_attrs_id) from jps_attrs;
Select max(logid) from jps_changelog;
```

If any of these return a higher number than that from the `sequence_name` query, increment that sequence to a higher value:

```
ALTER SEQUENCE xxx INCREMENT BY N
```

Where the `xxx` is the sequence being >= last_number

Make *N* greater than the value returned in the `jps-dn, jps_attrs`, and `jps_changelog` queries

Run the Reconfiguration Wizard again. .

Oracle Identity Manager Server Throws OutOfMemoryError

When you start the servers post upgrade, `OutOfMemoryError` is thrown.

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

```
[oim_server1] [NOTIFICATION] []
[oracle.iam.oimdataproviders.impl] [tid: [ACTIVE].ExecuteThread: '9' for
queue: 'weblogic.kernel.Default (self-tuning)'] [userId: xelsysadm] [ecid:
5679ce10-f0df-457f-88f1-6bc04e10aa13-000013b1,0] [APP: oim-runtime]
[partition-name: DOMAIN] [tenant-name: GLOBAL] [DSID:
0000Lg0PPYTBd5I_Ipt1if1OpGGi00000U] RM_DEBUG_PERF - 2017-03-24 06:09:51.087 -
search criteria = arg1 = (usr_key) EQUAL arg2 = (1)[[
 query = Select usr.usr_key, usr.usr_status  from usr where usr.usr_key = ?
 time = 1
]]
[2017-03-24T06:09:52.286-07:00] [oim_server1] [NOTIFICATION] []
[oracle.iam.oimdataproviders.impl] [tid: [ACTIVE].ExecuteThread: '9' for
queue: 'weblogic.kernel.Default (self-tuning)'] [userId: xelsysadm] [ecid:
5679ce10-f0df-457f-88f1-6bc04e10aa13-000013b1,0] [APP: oim-runtime]
[partition-name: DOMAIN] [tenant-name: GLOBAL] [DSID:
0000Lg0PPYTBd5I_Ipt1if1OpGGi00000U]
oracle.iam.oimdataproviders.impl.OIMUserDataProvider
[2017-03-24T06:11:52.171-07:00] [oim_server1] [ERROR] [ADFC-50018]
[oracle.adfinternal.controller.application.AdfcExceptionHandler] [tid:
[ACTIVE].ExecuteThread: '27' for queue: 'weblogic.kernel.Default
(self-tuning)'] [userId: xelsysadm] [ecid:
5679ce10-f0df-457f-88f1-6bc04e10aa13-000013e0,0] [APP:
oracle.iam.console.identity.self-service.ear] [partition-name: DOMAIN]
[tenant-name: GLOBAL] [DSID: 0000Lg0RtM9Bd5I_Ipt1if1OpGGi00000V] ADFc: No
exception handler was found for an application exception.[[
java.lang.OutOfMemoryError: GC overhead limit exceeded ]
```

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

1. Ensure that you set the following parameters in the `/etc/security/limits.conf` file, to the specified values:

   ```
   FUSION_USER_ACCOUNT soft nofile 32767
   FUSION_USER_ACCOUNT hard nofile 327679
   ```

2. Ensure that you set `UsePAM` to `Yes` in the `/etc/ssh/sshd_config` file.
3. Restart `sshd`.
4. Log out (or reboot) and log in to the system again.

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

Failure in UPDATE_WORKFLOW_POLICIES Post-Bootstrap Task

The `UPDATE_WORKFLOW_POLICIES` post-bootstrap task fails when you start the OIM Managed server after the upgrade.

The OIM Managed server displays the following error message:

```
Update WF policies started. Update SOA composite name from
default/DefaultRequestApproval!5.0 to default/DefaultRequestApproval!6.0>
<Apr 13, 2021 5:09:50,451 PM UTC> <Error> <OIM Authenticator> <BEA-000000>
<Authentication of user xelsysadm failed because of invalid password>
```

The OIM Managed server fails because the OIM administrator password is incorrect in the CSF keys.

Solution

Ensure that the OIM administrator (`xelsysadm`) password is same and correct in the following CSF keys:

Table A-1 OIM Managed Server CSF Keys

Sl. No	CSF Map	CSF Key
1.	`oracle.wsm.security`	`OIMAdmin`
2.	`oim`	`sysadmin`

To correct the password of the CSF keys:

1. Log in to the Oracle Enterprise Manager Console with the WebLogic administrator credentials.
2. From the **WebLogic Domain** drop-down, select **Security**, and then **Credentials**.
3. On the Credentials page, expand the **oim** CSF map, select the **sysadmin** CSF key, and then click the **Edit** icon to change the XELSYSADM credentials from the pop-up window.
4. Repeat Step 3 for the **OIMAdmin** CSF Key under **oracle.wsm.security** CSF Map.

MDS Customizations are Removed After You Restart the OIM Managed Server of an Upgraded Setup

If any MDS customizations are done after a successful upgrade to 14c (14.1.2.1.0) and if those customizations are lost after you restart the OIM Managed Server, you cannot recover the MDS changes. You have to do the MDS customizations again.

To avoid the repeated occurrence of this issue each time you restart the Managed Server, replace the existing 14c (14.1.2.1.0)_*ORACLE_HOME*>`/idm/server/apps/oim.ear/metadata.tar` file with the file that is present at the same location after you install the 14c (14.1.2.1.0) binaries, prior to the upgrade.

> **Note:**
>
> This issue is applicable only for MDS customizations that were made after the successful upgrade to 12c but lost after restarting the OIM Managed Server.

As part of the pre-upgrade tasks, after installing the 14c (14.1.2.1.0) binaries, you would have already taken a backup of the original 14c (14.1.2.1.0)_*ORACLE_HOME*>/idm/server/apps/oim.ear/metadata.tar file. See Backing Up the metadata.mar File Manually.

If the backup of the original file is not present after you install the binaries, you should install the 14c (14.1.2.1.0) binaries at any temporary location and extract the file.

For a HA setup, the original 14c (14.1.2.1.0)_*ORACLE_HOME*>/idm/server/apps/oim.ear/metadata.tar file is present on the secondary nodes where upgrade bootstrap was not executed.

OPatch Fails for not Finding the 'fuser' Command

OPatch fails when it is unable to locate the `fuser` command.

OPatch fails with the following error on the command line:

```
Verifying environment and performing prerequisite checks...
Prerequisite check "CheckActiveFilesAndExecutables" failed.
The details are:
Exception occured : fuser could not be located:
UtilSession failed: Prerequisite check "CheckActiveFilesAndExecutables" failed.
Log file location: <PATH>/fmw/cfgtoollogs/opatch/opatch20xx-0x-20_11-40-12AM_1.log
```

Following options are available to resolve this issue:

Pass argument for OPatch to ignore `fuser` and continue with patching:

1. Set the environment variable *OPATCH_NO_FUSER*=`true`. Setting this variable to "true" informs OPatch to skip the check for active executables.
2. Shut down the WebLogic instances.
3. Run the OPatch utility.

Set a temporary `fuser`:

1. Set `/tmp` in your PATH.
2. Create an empty file named "`fuser`".
3. Shut down the WebLogic instances.
4. Run the OPatch utility.

Install the 'fuser' utility:

1. Install the 'fuser' utility on the machine (contact your OS Admin).
2. Ensure that 'fuser' is located under `/sbin/fuser` or `/bin/fuser`.
3. Shut down the WebLogic instances.
4. Run the OPatch utility.

Administration Server Has a Slow Start After the Upgrade

The Administration Server experiences a slow start after the upgrade.

The thread dump displays the following information:

```
[ACTIVE] ExecuteThread: '5' for queue: 'weblogic.kernel.Default
(self-tuning)'" #76 daemon prio=5 os_prio=0 tid=0x00007f4fcc008000 nid=0x20c6
runnable [0x00007f4fbc2d6000]
  java.lang.Thread.State: RUNNABLE
at java.io.FileInputStream.readBytes(Native Method)
at java.io.FileInputStream.read(FileInputStream.java:255)
at sun.security.provider.NativePRNG$RandomIO.readFully(NativePRNG.java:424)
at
sun.security.provider.NativePRNG$RandomIO.implGenerateSeed(NativePRNG.java:441
)
- locked <0x0000000640b92be8> (a java.lang.Object)
at sun.security.provider.NativePRNG$RandomIO.access$500(NativePRNG.java:331)
at sun.security.provider.NativePRNG.engineGenerateSeed(NativePRNG.java:226)
at java.security.SecureRandom.generateSeed(SecureRandom.java:546)
at
com.bea.security.utils.random.AbstractRandomData.ensureInittedAndSeeded(Abstra
ctRandomData.java:92)
- locked <0x000000075b7af6b8> (a
com.bea.security.utils.random.SecureRandomData)
at
com.bea.security.utils.random.AbstractRandomData.getRandomLong(AbstractRandomD
ata.java:117)
- locked <0x000000075b7af6b8> (a
com.bea.security.utils.random.SecureRandomData)
```

To resolve this issue, set the `-Djava.security.egd=file:/dev/./urandom` parameter in the `JAVA_OPTIONS` section of the `setDomainEnv.sh/cmd` file and restart the server.

NPE Encountered on Starting OIM Server After Running the Upgrade Assistant

A Null Pointer Exception (NPE) is encountered when starting the OIM server after running the Upgrade Assistant for upgrading the domain configuration.

The OIM server fails to start and displays the following error message:

```
Exception[[
java.lang.NullPointerException
   at
oracle.iam.rcu.LoadTemplateDataLogger.writeLog(LoadTemplateDataLogger.java:31)

   at
oracle.iam.rcu.LoadTemplates.loadAllTempalteImplementation(LoadTemplates.java:
113)
   at oracle.iam.rcu.LoadTemplates.loadAllTemplates(LoadTemplates.java:168)
   at
oracle.iam.OIMPostConfigManager.config.OIMConfigManager.seedNotificationTempla
te(OIMConfigManager.java:2866)
   at
oracle.iam.OIMPostConfigManager.config.OIMConfigManager.executeAndRegisterTask
(OIMConfigManager.java:1754)
   at
```

```
oracle.iam.OIMPostConfigManager.config.OIMConfigManager.configureOIM(OIMConfig
Manager.java:1558)
   at
oracle.iam.OIMPostConfigManager.config.OIMConfigManager.doExecute(OIMConfigMan
ager.java:1179)
   at
oracle.iam.OIMPostConfigManager.appListener.BootStrapListener.preStart(BootStr
apListener.java:134)
```

To resolve this error, you should include `/idm` in the value of *ORACLE_HOME* in the `setDomainEnv.sh` file.

For example: `/u01/oracle/product/`*`ORACLE_HOME`*`/idm`

OIM Bootstrap Fails Due to the Presence of Custom Application JARs

If there are any custom developed libraries or JARs placed inside the *OIM_HOME*, the OIM bootstrap fails during the upgrade to Oracle Identity Manager 14c (14.1.2.1.0).

The failure results in an error message similar to the following:

```
<Server state changed to FORCE_SHUTTING_DOWN.>
<Nov 19, 2020 4:04:50,356 PM EST> <Notice> <Log Management> <BEA-170037> <The
log monitoring service timer has been stopped.>
<Nov 19, 2020 4:06:16,377 PM EST> <Warning> <JMX> <BEA-149513> <JMX Connector
Server stopped at
service:jmx:iiop://idmoimtl3.chop.edu:14000/jndi/weblogic.management.mbeanserv
ers.runtime.>
<Nov 19, 2020 4:15:43,045 PM EST> <Error> <netuix> <BEA-423142> <The control
com.bea.netuix.servlets.controls.layout.Layout could not be rendered properly
due to the following error:>
<Nov 19, 2020 4:15:44,356 PM EST> <Warning> <Socket> <BEA-000449> <Closing
the socket, as no data read from it on 10.250.116.181:54,532 during the
configured idle timeout of 5 seconds.>
<Nov 19, 2020 4:17:57,525 PM EST> <Warning> <J2EE> <BEA-160188> <Unresolved
application library references, for application
oracle.iam.console.identity.self-service.ear, defined in
weblogic-application.xml: [Extension-Name: oracle.iam.ui.model, exact-match:
false].>
<Nov 19, 2020 4:17:57,810 PM EST> <Warning> <J2EE> <BEA-160188> <Unresolved
WebApp library references defined in weblogic.xml, of module
'oracle.iam.console.identity.self-service.war' [Extension-Name:
oracle.iam.ui.view, exact-match: false], [Extension-Name:
oracle.iam.ui.oia-view, exact-match: false], [Extension-Name:
oracle.iam.ui.custom, exact-match: false], [Extension-Name:
oracle.idm.msm.ui.library, exact-match: false].>
java.lang.ClassNotFoundException:
oracle.iam.ui.platform.view.backing.SkinBean at
weblogic.utils.classloaders.GenericClassLoader.findLocalClass(GenericClassLoad
er.java:1029) at
weblogic.utils.classloaders.GenericClassLoader.findClass(GenericClassLoader.ja
va:990)  at
weblogic.utils.classloaders.GenericClassLoader.doFindClass(GenericClassLoader.
java:611) at
```

```
weblogic.utils.classloaders.GenericClassLoader.loadClass(GenericClassLoader.ja
va:543) at
weblogic.servlet.internal.AnnotationProcessingManager.processAnnotations(Annot
ationProcessingManager.java:105) at
weblogic.servlet.tools.WARModule.processAnnotations(WARModule.java:513) at
weblogic.servlet.tools.WARModule.processAnnotations(WARModule.java:605) at
weblogic.servlet.tools.WARModule.merge(WARModule.java:553) at
weblogic.application.compiler.ToolsModuleWrapper.merge(ToolsModuleWrapper.java
:96) at
weblogic.application.utils.CustomModuleManager.merge(CustomModuleManager.java:
78) at
weblogic.application.compiler.flow.MergeModuleFlow.compile(MergeModuleFlow.jav
a:38) at
weblogic.application.compiler.FlowDriver$FlowStateChange.next(FlowDriver.java:
70) at
weblogic.application.utils.StateMachineDriver.nextState(StateMachineDriver.jav
a:45) at
weblogic.application.compiler.FlowDriver.nextState(FlowDriver.java:37)
weblogic.application.compiler.flow.AppMergerFlow.mergeInput(AppMergerFlow.java
:75)at
weblogic.application.compiler.flow.AppMergerFlow.compile(AppMergerFlow.java:40
) at
weblogic.application.compiler.FlowDriver$FlowStateChange.next(FlowDriver.java:
70) at
weblogic.application.utils.StateMachineDriver.nextState(StateMachineDriver.jav
a:45) at
weblogic.application.compiler.FlowDriver.nextState(FlowDriver.java:37) at
weblogic.application.compiler.AppMerge.runBody(AppMerge.java:168) at
weblogic.utils.compiler.Tool.run(Tool.java:159) at
weblogic.utils.compiler.Tool.run(Tool.java:116) at
weblogic.application.compiler.AppMerge.merge(AppMerge.java:198) at
weblogic.deploy.api.internal.utils.AppMerger.merge(AppMerger.java:94)at
weblogic.deploy.api.internal.utils.AppMerger.getMergedApp(AppMerger.java:58)
at
weblogic.deploy.api.model.internal.WebLogicDeployableObjectFactoryImpl.createD
eployableObject(WebLogicDeployableObjectFactoryImpl.java:186) at
weblogic.deploy.api.model.internal.WebLogicDeployableObjectFactoryImpl.createD
eployableObject(WebLogicDeployableObjectFactoryImpl.java:167)at
com.bea.console.utils.DeploymentConfigurationHelper$1.execute(DeploymentConfig
urationHelper.java:860) at
com.bea.console.utils.DeploymentUtils.runDeploymentAction(DeploymentUtils.java
:5690) at
com.bea.console.utils.DeploymentConfigurationHelper.initDeploymentConfiguratio
n(DeploymentConfigurationHelper.java:848) at
com.bea.console.utils.DeploymentConfigurationHelper.completeInitialization(Dep
loymentConfigurationHelper.java:444) at
com.bea.console.utils.DeploymentConfigurationManager.getDeploymentConfiguratio
n(DeploymentConfigurationManager.java:151) at
com.bea.console.utils.DeploymentConfigurationManager.getDeploymentConfiguratio
n(DeploymentConfigurationManager.java:104) at
```

To resolve this issue, Oracle recommends not to keep the custom-developed JARs or libraries inside OIM_HOME to avoid file system dependencies. The file system dependencies add an overhead of maintaining such custom libraries during the out-of-place Oracle Home upgrades because such custom JARs remain in the old Oracle Home (Oracle Home before the upgrade process).

To avoid such issues, you should upload the custom libraries to the database. If the custom library is in the OIM plug-in compressed (.zip) format, register them using the plug-in utility. If the custom library is a JAR, upload the same to the database using the Upload JAR Utility.

If for some reason, you do not want to follow the above recommendations, you can manually copy the custom-developed JARs from the old to the new Oracle home, in the appropriate location.

Incorrect Links in Password Reset Emails

The OIG system generated password reset email has links in the `applewebdata://<ANY_RANDOM_GUID>/null` format, which is incorrect.

To resolve this issue, update the `OIMExternalFrontEndURL` parameter with the correct value in the `Discovery` MBean of OIM by completing the following steps:

1. Log in to the Enterprise Manager Console.
2. Navigate to **System MBean Browser**.
3. Under **Application Defined MBeans**, navigate to **oracle.iam**, select **Server <server>**, click **Application:oim**, click **XMLConfig**, select `Config`, select **XMLConfig.DiscoveryConfig**, and then click **Discovery**.
4. Update the `OIMExternalFrontEndURL` parameter with the appropriate value. This parameter should not be empty.

B

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.

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.
- Updating the JDK Location in an Existing Domain Home
  You must search the references to the current JDK, for example 1.8.0_191 manually, and replace those instances with the location of the new JDK.

Updating the JDK Location in an Existing Oracle Home

The `getProperty.sh|cmd` script displays the value of a variable, such as JAVA_HOME, from the .globalEnv.properties file. The `setProperty.sh|cmd` script is used to set the value of variables, such as OLD_JAVA_HOME or JAVA_HOME that contain the locations of old and new JDKs in the .globalEnv.properties file.

The `getProperty.sh|cmd` and `setProperty.sh|cmd` scripts are located in the following location:

(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 1.8.0_191.

To update the JDK location in the .globalEnv.properties file:

1. Use the `getProperty.sh|cmd` script to display the path of the current JDK from the JAVA_HOME variable. For example:

   (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, for example 1.8.0_191 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