

# Oracle® Fusion Middleware

## Upgrading Oracle Identity Manager



12c (12.2.1.3.0)

E95500-05

July 2020

The Oracle logo, consisting of the word "ORACLE" in white, uppercase, sans-serif font, centered within a solid red square.

ORACLE®

Oracle Fusion Middleware Upgrading Oracle Identity Manager, 12c (12.2.1.3.0)

E95500-05

Copyright © 2017, 2020, Oracle and/or its affiliates.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or "commercial computer software documentation" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

# Contents

## Preface

---

Audience	viii
Documentation Accessibility	viii
Related Documents	viii
Conventions	ix

## 1 Introduction to Upgrading Oracle Identity Manager to 12c (12.2.1.3.0)

---

About the Starting Points for a Oracle Identity Manager Upgrade	1-2
About the Oracle Identity Manager Upgrade Scenarios	1-2
About the New Features for Oracle Identity Manager 12c	1-3
About Upgrade Restrictions	1-3
Terminology Used in this Guide	1-4
How to Use This Guide	1-5

## 2 Pre-Upgrade Requirements

---

Oracle Fusion Middleware Pre-Upgrade Checklist	2-2
Creating a Complete Backup	2-3
Backing Up the Schema Version Registry Table	2-4
Maintaining Customized Domain and Environment Settings	2-4
Cloning Your Production Environment for Testing	2-5
Verifying Certification and System Requirements	2-6
Verify Your Environment Meets Certification Requirements	2-7
Verify System Requirements and Specifications	2-7
Migrating from a 32-Bit to a 64-Bit Operating System	2-8
Verify That the Database Hosting Oracle Fusion Middleware is Supported	2-11
Verify That the JDK Is Certified for This Release of Oracle Fusion Middleware	2-11
Updating Policy Files when Using Enhanced Encryption (AES 256)	2-12
Purging Unused Data	2-12
Creating a Non-SYSDBA User to Run the Upgrade Assistant	2-13
Identifying Existing Schemas Available for Upgrade	2-15

Updating Database Parameters for Oracle Identity Manager	2-16
Updating Connectors for Oracle Identity Manager	2-17
Generating and Analyzing Pre-Upgrade Report for Oracle Identity Manager	2-17
Obtaining the Pre-Upgrade Report Utility	2-18
Generating the Pre-Upgrade Report	2-18
Analyzing the Pre-Upgrade Report	2-20

### 3 Upgrading Oracle Identity Manager Single Node Environments

---

About the Oracle Identity Manager Single Node Upgrade Process	3-2
Completing the Pre-Upgrade Tasks for Oracle Identity Manager	3-4
Updating Server Wallets to Remove MD5 Algorithm	3-5
Updating DB Wallets to Remove MD5 Algorithm (For SSL Enabled Setup)	3-8
Verifying the Memory Settings	3-10
Opening the Non-SSL Ports for SSL Enabled Setup	3-11
Installing Product Distributions	3-11
Running a Pre-Upgrade Readiness Check	3-14
About Running a Pre-Upgrade Readiness Check	3-14
Starting the Upgrade Assistant in Readiness Mode	3-15
Upgrade Assistant Parameters	3-15
Performing a Readiness Check with the Upgrade Assistant	3-17
Understanding the Readiness Report	3-19
Creating the Required 12c Schemas Using RCU (Optional)	3-23
Tuning Database Parameters for Oracle Identity Manager	3-27
Stopping Servers and Processes	3-27
Upgrading Product Schemas	3-29
Identifying Existing Schemas Available for Upgrade	3-30
Starting the Upgrade Assistant	3-31
Upgrading Oracle Identity Manager Schemas Using the Upgrade Assistant	3-32
Verifying the Schema Upgrade	3-36
About Reconfiguring the Domain	3-37
Backing Up the Domain	3-39
Starting the Reconfiguration Wizard	3-40
Reconfiguring the Oracle Identity Manager Domain	3-41
Upgrading Domain Component Configurations	3-43
Starting the Upgrade Assistant	3-44
Upgrade Assistant Parameters	3-45
Upgrading Oracle Identity Manager Domain Component Configurations	3-47
Verifying the Domain-Specific-Component Configurations Upgrade	3-49
Post-Upgrade Task	3-50
Upgrading SOA Composites	3-50

Copying folders to the 12c Oracle Home	3-51
Starting the Servers	3-52
Starting Servers and Processes	3-53
Upgrading Oracle Identity Manager Design Console	3-54
Completing the Post-Upgrade Tasks for SSL Enabled Setup	3-54
Installing Standalone Oracle BI Publisher	3-55
Tuning Application Module for User Interface	3-56

## 4 Upgrading Oracle Identity Manager Highly Available Environments

---

About the Oracle Identity Manager Multinode Upgrade Process	4-2
Completing the Pre-Upgrade Tasks for Oracle Identity Manager	4-5
Updating Server Wallets to Remove MD5 Algorithm	4-6
Updating DB Wallets to Remove MD5 Algorithm (For SSL Enabled Setup)	4-9
Verifying the Memory Settings	4-11
Opening the Non-SSL Ports for SSL Enabled Setup	4-12
Creating 12c Oracle Home Folder on OIMHOST1 and OIMHOST2	4-12
Installing Product Distributions on OIMHOST1 and OIMHOST2	4-12
Installing Product Distributions	4-13
Running a Pre-Upgrade Readiness Check	4-16
About Running a Pre-Upgrade Readiness Check	4-16
Starting the Upgrade Assistant in Readiness Mode	4-17
Upgrade Assistant Parameters	4-17
Performing a Readiness Check with the Upgrade Assistant	4-19
Understanding the Readiness Report	4-21
Creating the Required 12c Schemas Using RCU (Optional)	4-25
Stopping Servers and Processes	4-29
Upgrading Schemas on OIMHOST1	4-31
Upgrading Product Schemas	4-31
Identifying Existing Schemas Available for Upgrade	4-32
Starting the Upgrade Assistant	4-33
Upgrading Oracle Identity Manager Schemas Using the Upgrade Assistant	4-34
Verifying the Schema Upgrade	4-38
Reconfiguring the Domain on OIMHOST1	4-39
About Reconfiguring the Domain	4-39
Backing Up the Domain	4-41
Starting the Reconfiguration Wizard	4-42
Reconfiguring the Oracle Identity Manager Domain	4-43
Upgrading Domain Component Configurations on OIMHOST1	4-46
Upgrading Domain Component Configurations	4-46
Starting the Upgrade Assistant	4-46

Upgrading Oracle Identity Manager Domain Component Configurations	4-49
Verifying the Domain-Specific-Component Configurations Upgrade	4-52
Starting the Servers	4-52
Replicating the Domain Configurations on each OIMHOST	4-53
Copying oracle.iam.ui.custom-dev-starter-pack.war to the 12c (12.2.1.3.0) Oracle Home	4-54
Starting the Servers on OIMHOST1 and OIMHOST2	4-54
Starting Servers and Processes	4-55
Configuring Oracle HTTP Servers to Front End OIM, and SOA Managed Servers	4-56
Upgrading Oracle Identity Manager Design Console	4-60
Completing the Post-Upgrade Tasks for SSL Enabled Setup	4-60
Installing Standalone Oracle BI Publisher	4-61

## 5 Upgrading OIM-OAM Integrated Environments set up Manually

---

About the OIM-OAM Integrated HA Topology Set Up Manually	5-1
Supported Starting Points for Integrated HA Upgrade	5-3
Roadmap for Upgrading OIM-OAM Integrated Highly Available Environments Set Up Manually	5-3
Enabling Oracle Identity Governance 12c to Work With Oracle Access Manager 11g	5-4

## 6 Upgrading OIM-OAM Integrated Environments set up Using Life Cycle Management Tool

---

About the OIM-OAM Integrated HA Topology Set Up Using LCM Tool	6-1
Supported Starting Points	6-3
Roadmap for Upgrading OIM-OAM Integrated Environments set up Using Life Cycle Management Tool	6-3

## 7 Troubleshooting the Oracle Identity Manager Upgrade

---

WebLogic Server is Not in the Running Status	7-3
Upgrading Product Schemas Error: OIMR2PS2_OIM.PK_POP	7-5
Upgrading Product Schemas Error: OIMR2PS2_OIM.UK_ENTITY_TYPE	7-6
KeystoreService Exception in the Logs After Reconfiguring the OIM Domain	7-6
Warning when Generating the Pre-Upgrade Report for OIM	7-8
Domain Reconfiguration Error	7-8
OIM Bootstrap for DEPLOYSOACOMPOSITES Task Fails After Upgrade	7-10
Authorization Policy Merge Issue	7-12
MAR Update or Metadata Merge Issue	7-13
Error When Opening ADF DI Excel Sheet After Upgrade	7-14

Compilation Error When Starting the SOA Server After Upgrade	7-15
Warning in Oracle Identity Manager Server Logs After Upgrade	7-16
Default Challenges Questions are not Updated After Upgrade	7-16
OPSS Processing Error When Reconfiguring the Domain	7-17
EditFailedException When Releasing Configuration From WebLogic Console	7-17
OIM Application Deployment Fails Intermittently	7-18
soa-infra Application is in 'Prepared' State Post Upgrade	7-18
Oracle Identity Manager Server Throws OutOfMemoryError	7-18
SOA Fails to Join Coherence Cluster During the First Start After Upgrade	7-19
LDAP User Create and Update Reconciliation Job Fails	7-20
BI Managed Server is Seen on WebLogic Console After Upgrade	7-21
Empty Pages or Panels After Upgrade	7-21

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

---

About Updating the JDK Location After Installing an Oracle Fusion Middleware Product	A-1
Updating the JDK Location in an Existing Oracle Home	A-3
Updating the JDK Location in an Existing Domain Home	A-3

# Preface

This document describes how to upgrade an existing Oracle Identity Manager environment to 12c (12.2.1.3.0).

- [Audience](#)  
Identify the target audience for your book and learn more about this document intended for.
- [Documentation Accessibility](#)
- [Related Documents](#)
- [Conventions](#)  
Learn about the conventions used in this document.

## Audience

Identify the target audience for your book and learn more about this document intended for.

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.

## 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
<b>boldface</b>	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	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 12c (12.2.1.3.0)

Before you begin, review all introductory information to understand the standard upgrade topologies and upgrade paths for Oracle Identity Manager 12c (12.2.1.3.0).

### Note:

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

### Note:

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 12c \(12.2.1.3.0\)](#)
- [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](#)  
You can upgrade to Oracle Identity Manager 12c (12.2.1.3.0) from a supported 11g release.
- [About the Oracle Identity Manager Upgrade Scenarios](#)  
The steps to upgrade Oracle Identity Manager to 12c (12.2.1.3.0) depend on the existing 11g Release 2 (11.1.2.3.0) production topology.
- [About the New Features for Oracle Identity Manager 12c](#)  
Several changes have been made to Oracle Identity Manager between 11g and 12c.
- [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

You can upgrade to Oracle Identity Manager 12c (12.2.1.3.0) from a supported 11g release.

If you are not using the 11.1.2.3.0 version of Oracle Identity Manager, you must upgrade to 11.1.2.3.0 before you move to 12c (12.2.1.3.0).

For information about upgrading Oracle Identity Manager to 11g Release 2 (11.1.2.3.0), see [Introduction to Oracle Identity and Access Management Upgrade](#) in the *Upgrade Guide for Oracle Identity and Access Management* for 11g Release 2 (11.1.2.3.0).

The upgrade procedures in this guide explain how to upgrade an existing Oracle Identity Manager 11g domain to Oracle Identity Manager 12c (12.2.1.3.0). 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 12c (12.2.1.3.0) depend on the existing 11g Release 2 (11.1.2.3.0) production topology.

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

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 [Planning an Upgrade to Oracle Fusion Middleware 12c \(12.2.1.3.0\)](#) 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](#)
- [Oracle Identity Manager and Oracle Access Manager integrated environments that are set up manually in 11.1.2.3.0](#)
- [Oracle Identity Manager and Oracle Access Manager integrated environments that are set up using Life Cycle Management \(LCM\) tool in 11.1.2.3.0](#)

 **Note:**

If you are using Oracle Access Manager Mobile and Social, do NOT upgrade to 12c (12.2.1.3.0). Contact Oracle support for more details on the upgrade path for Mobile and Social.

## About the New Features for Oracle Identity Manager 12c

Several changes have been made to Oracle Identity Manager between 11g and 12c.

To understand what's new in general in Oracle Fusion Middleware 12c, 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 12c*.

For more information about Oracle Identity Governance 12c (12.2.1.3.0), see the following topics in *Administering Oracle Identity Governance*:

- [New and Changed Features for 12c \(12.2.1.3.0\)](#)
- [What is Oracle Identity Governance?](#)
- [Features Not Supported in Oracle Identity Governance 12c \(12.2.1.3.0\)](#)
- [What are the Different Modes of Oracle Identity Governance?](#)
- [Oracle Identity Governance Sizing Guide 12c \(12.2.1.3.0\)](#)
- [Oracle Fusion Middleware 12c \(12.2.1.3.0\) Certification Matrix](#)

## 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 12c (12.2.1.0) components are generally interoperable with other 12c (12.2.1.0) components.

### 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 12.2.1.3.0, see *Products and Features Available in Oracle Fusion Middleware 12c (12.2.1.3.0) 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/ jdk1.8.0_131	Environment variable that points to the Java JDK home directory.
Database host	examplehost.exampledomain	Name and domain of the host where the database is running.
Database port	1521	Port number that the database listens on. The default Oracle database listen port is 1521.
Database service name	orcl.exampledomain	Oracle databases require a unique service name. The default service name is orcl.
DBA username	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	/home/Oracle/product/ ORACLE_HOME	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.  <b>Note:</b> 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.

**Table 1-1 (Cont.) Terminology**

Information	Example Value	Description
Administrator user password	<i>&lt;admin_password&gt;</i>	Password of the user with Oracle WebLogic Server administration privileges.
RCU	<i>ORACLE_HOME/ oracle_common/bin</i>	Path to the Repository Creation Utility (RCU).
RCU schema prefix	<i>oim</i>	Prefix for names of database schemas used by Oracle Identity Manager.
RCU schema password	<i>&lt;rcu_password&gt;</i>	Password for the database schemas used by Oracle Identity Manager.
Configuration utility	<i>ORACLE_HOME/ oracle_common/ common/bin</i>	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 11.1.2.3.0 deployment, refer to the respective topics for upgrading Oracle Identity Manager to 12c (12.2.1.3.0):

- **Single Node Environments:**

For upgrading single node Oracle Identity Manager (OIM) setup, see [Upgrading Oracle Identity Manager Single Node Environments](#).

- **Multi-node or Highly Available Environments:**

For upgrading multi-node Oracle Identity Manager setup, see [Upgrading Oracle Identity Manager Highly Available Environments](#).

- **OIM-OAM Integrated Highly Available Environments:**

- For upgrading OIM-OAM integrated highly available deployment, that was set up manually in 11g, see [Upgrading OIM-OAM Integrated Environments set up Manually](#).
- For upgrading OIM-OAM integrated highly available deployment, that was set up using Life Cycle Management (LCM) tool in 11g, see [Upgrading OIM-OAM Integrated Environments set up Using Life Cycle Management Tool](#).

 **Note:**

Before you begin the upgrade, ensure that you review the [Pre-Upgrade Requirements](#) and perform necessary pre-upgrade tasks.

# 2

## Pre-Upgrade Requirements

Before you begin to upgrade Oracle Identity Manager 12c (12.2.1.3.0), you must perform pre-upgrade tasks such as backing up, cloning 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, Middleware home, and databases that host your Oracle Fusion Middleware schemas.
- [Cloning Your Production Environment for Testing](#)  
Create a copy of your actual production environment, upgrade the cloned environment, verify that the upgraded components work as expected, and only then upgrade your production environment.
- [Verifying Certification and System Requirements](#)  
Review the certification matrix and system requirements documents to verify that your environment meets the necessary requirements for installation.
- [Updating Policy Files when Using Enhanced Encryption \(AES 256\)](#)  
If you plan to use enhanced encryption, such as Advanced Encryption Standard (AES 256), in your upgraded environment, Oracle recommends that you apply the latest required policy files to the JDK before you upgrade.
- [Purging Unused Data](#)  
Purging unused data and maintaining a purging methodology before an upgrade can optimize the upgrade process.
- [Creating a Non-SYSDBA User to Run the Upgrade Assistant](#)  
Oracle recommends that you create a non-SYSDBA user called `FMW` to run the Upgrade Assistant. This user has the privileges required to modify schemas, but does not have full administrator privileges.
- [Identifying Existing Schemas Available for Upgrade](#)  
This optional task enables you to review the list of available schemas before you begin the upgrade by querying the schema version registry. The registry contains schema information such as version number, component name and ID, date of creation and modification, and custom prefix.
- [Updating Database Parameters for Oracle Identity Manager](#)  
You need to verify and update a few database parameters before upgrading the Oracle Identity Manager to 12c (12.2.1.3.0).
- [Updating Connectors for Oracle Identity Manager](#)  
Update the existing connectors if they are not supported for Oracle Identity Manager 12c (12.2.1.3.0).

- [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.

## 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.

Ensure that Oracle Identity Manager and Oracle Access Manager are in different domains. If they are in the same domain, then you need to separate them into multiple domains. For more information, see [Separating Oracle Identity Management Applications Into Multiple Domains](#).

**Table 2-1 Tasks to Perform Before You Upgrade to Oracle Fusion Middleware 12c**

Task	Description
<p><b>Required</b></p> <p>Create a complete backup of your existing environment.</p>	<p>Back up all system-critical files, including the Oracle home, Middleware home, and databases 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.</p> <p>See <a href="#">Creating a Complete Backup</a>.</p> <ul style="list-style-type: none"> <li>• Ensure that your backup includes the schema version registry table. See <a href="#">Backing Up the Schema Version Registry Table</a>.</li> <li>• 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 <a href="#">Maintaining Customized Domain and Environment Settings</a>.</li> </ul>
<p><b>Optional</b></p> <p>Clone your production environment to use as an upgrade testing platform.</p>	<p>In addition to creating a complete backup of your system files, Oracle strongly recommends that you clone your production environment. This environment can be used to test the upgrade.</p> <p>See <a href="#">Cloning Your Production Environment for Testing</a>.</p>



Table 2-1 (Cont.) Tasks to Perform Before You Upgrade to Oracle Fusion Middleware 12c

Task	Description
<p><b>Required</b></p> <p>Verify that you are installing and upgrading your product on a supported hardware and software configuration.</p> <p><b>Caution:</b> 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.</p>	<p>Verify that your hardware and software configurations (including operating systems) are supported by the latest certifications and requirements. Also make sure to use a supported JDK version before you install the 12c product distributions.</p> <p>Oracle recommends that: you verify this information right before you start the upgrade as the certification requirements are frequently updated.</p> <p><b>Note:</b></p> <ul style="list-style-type: none"> <li>• Ensure that you have applied the latest patches to your components before you upgrade.</li> <li>• 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.</li> </ul> <p>See <a href="#">Verifying Certification and System Requirements</a>.</p>
<p><b>Required for 32-bit Operating Systems Only</b></p> <p>Migrate to a 64-bit operating system before you upgrade.</p>	<p>This is required only if you are currently running an unsupported 32-bit operating system.</p> <p>See <a href="#">Migrating from a 32-Bit to a 64-Bit Operating System</a>.</p>
<p><b>Optional</b></p> <p>Update security policy files if you are using enhanced encryption (AES 256).</p>	<p>Some of the security algorithms used in Fusion Middleware 12c require additional policy files for the JDK.</p> <p>If you plan to use enhanced encryption, such as AES 256, Oracle recommends that you apply the latest required policy files to the JDK before you upgrade.</p> <p>See <a href="#">Updating Policy Files when Using Enhanced Encryption (AES 256)</a>.</p>
<p><b>Optional</b></p> <p>Purge any outdated or unused data before you upgrade.</p>	<p>To optimize performance, Oracle strongly recommends that you purge data and objects that will not be used in the upgraded environment.</p> <p>See <a href="#">Purging Unused Data</a>.</p>
<p><b>Optional</b></p> <p>Create a Non-SYSDBA user to run the Upgrade Assistant.</p>	<p>Oracle recommends that you create the FMW user to run Upgrade Assistant. User FMW can run the Upgrade Assistant without system administration privileges.</p> <p>See <a href="#">Creating a Non-SYSDBA User to Run the Upgrade Assistant</a></p>
<p><b>Required</b></p> <p>Updating Connectors for Oracle Identity Manager.</p>	<p>See <a href="#">Updating Connectors for Oracle Identity Manager</a>.</p>

## Creating a Complete Backup

Before you start an upgrade, back up all system-critical files, including the Oracle home, Middleware 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 12c in *Planning an Upgrade of Oracle Fusion Middleware*
- [Oracle Database Documentation](#) for information about upgrading to Oracle Database 18c and 19c.
- [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, domain upgrade, and reconfiguration operations. Save your customized files to a shared library location so that you can continue to use them after the upgrade.

## 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.

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, domain upgrade, and reconfiguration operations.

Save your customized files to a shared library location so that you can continue to use them after the upgrade.

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. To maintain your custom domain-level environment settings, Oracle recommends that you create a separate file to store the custom domain information before you upgrade, instead of modifying the scripts directly.

For example, if you want to customize server startup parameters that apply to all servers in a domain, you can create a file called `setUserOverrides.cmd` (Windows) or `setUserOverrides.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.

The following example illustrates startup customizations in a `setUserOverrides` file:

```
# add custom libraries to the WebLogic Server system classpath
if [ "${POST_CLASSPATH}" != "" ] ; then
    POST_CLASSPATH="${POST_CLASSPATH}${CLASSPATHSEP}${HOME}/foo/
fooBar.jar"
    export POST_CLASSPATH
else
    POST_CLASSPATH="${HOME}/foo/fooBar.jar"
    export POST_CLASSPATH
fi

# specify additional java command-line options for servers
JAVA_OPTIONS="${JAVA_OPTIONS} -Dcustom.property.key=custom.value"
```

If the `setUserOverrides` file exists during a server startup, the file is included in the startup sequence and any overrides contained within this file take effect. You must store the `setUserOverrides` file in the `DOMAIN_HOME/bin` directory.

 **Note:**

If you are unable to create the `setUserOverrides` script before an upgrade, you need to reapply your settings as described in *Re-apply Customizations to Startup Scripts* in *Upgrading Oracle WebLogic Server*.

## Cloning Your Production Environment for Testing

Create a copy of your actual production environment, upgrade the cloned environment, verify that the upgraded components work as expected, and only then upgrade your production environment.

Cloning your production environment for testing is recommended, but not required.

Upgrades cannot be reversed. In most cases, if an error occurs, you must stop the upgrade and restore the entire environment from backup and begin the upgrade

process from the beginning. Identifying potential upgrade issues in a development environment can eliminate unnecessary downtime.

 **Note:**

It is beyond the scope of this document to describe the cloning procedures for all components and operating systems. Cloning procedures are component and operating system-specific. At a high level, you install the pre-upgrade version of your component domain on a test machine, create the required schemas using the Repository Creation Utility (RCU), and perform the upgrade.

Additional benefits of running an upgrade in a cloned production environment include the following:

- Uncover and correct any upgrade issues.
- Practice completing an end-to-end upgrade.
- Understand the upgrade performance and how purge scripts can help.
- Understand the time required to complete the upgrade.
- Understand the database resource usage (such as temporary tablespace; Program Global Area (PGA) , and so on).

 **Note:**

You can run the pre-upgrade Readiness Check on the cloned production environment to help identify potential upgrade issues with your data, but you must perform a complete test upgrade on a cloned environment to ensure a successful upgrade.

For more information, see [Doc ID 2621548.1](#).

## Verifying Certification and System Requirements

Review the certification matrix and system requirements documents to verify that your environment meets the necessary requirements for installation.

 **Note:**

When checking the certification, system requirements, and interoperability information, be sure to check specifically for any 32-bit or 64-bit system requirements. It is important for you to download software specifically designed for the 32-bit or 64-bit environment, explicitly.

- [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 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 12c.
- [Verify That the JDK Is Certified for This Release of Oracle Fusion Middleware](#)  
At the time this document was published, the certified JDK for 12c (12.2.1.3.0) was 1.8.0\_131.

## 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 12c (12.2.1.3.0).

## Verify System Requirements and Specifications

It is important 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 certification are met. For example, if the Certification Matrix for 12c (12.2.1.3.0) indicates that your product is certified for installation on 64-Bit Oracle Linux 7, verify that your Oracle Linux 7 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 Technical Resources.

### Note:

When you install the Oracle Fusion Middleware Release 12c 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 12c files and directories.

If you are running a 32-bit environment, you will need to perform an additional set of steps:

- [Migrating from a 32-Bit to a 64-Bit Operating System](#)  
If you have a 32-bit operating system, then you must migrate your 32-bit environment to a 64-bit software environment before you upgrade.

## Migrating from a 32-Bit to a 64-Bit Operating System

If you have a 32-bit operating system, then you must migrate your 32-bit environment to a 64-bit software environment before you upgrade.

Make sure to validate the migration to ensure all your Oracle Fusion Middleware 11g software is working properly on the 64-bit machine, and only then perform the upgrade to Oracle Fusion Middleware 12c.

In these tasks, *host* refers to the 32-bit source machine and *target* refers to the new 64-bit target machine.



### Note:

These steps assume that your database is located on a separate host and will not be moved.

Upgrading an operating system typically involves the following:



### Caution:

These steps are provided as an example of the operating system upgrade process and may or may not include all of the procedures you must perform to update your specific operating system. Consult your operating system's upgrade documentation for more information.

- [Procure the Hardware That Supports the Upgrade's 64-bit Software Requirement](#)  
Make sure that you have supported target hardware in place before you begin the upgrade process.
- [Stop All Processes](#)  
Before upgrading, you must stop all processes, including Managed Servers, the Administration Server, and Node Manager, if they are started on the host.
- [Back Up All Files from the 32-bit Host Machine](#)  
Make sure that you have created a complete backup of your entire 11g deployment before you begin the upgrade process. These files can be used if there is an issue during the migration and you have to restart the process.
- [Set Up the Target 64-bit Machine with the 11g Host Name and IP Address](#)  
The host name and IP address of the target machine must be made identical to the host. This requires you to change the IP address and name of the source machine or decommission the source machine to avoid conflicts in the network.
- [Restore the 11g Backup from 32-bit Host to 64-bit Host](#)  
Restore the files you backed from the 32-bit host using the same directory structure that was used in 11g. The directory structure on the target machine must be identical to the structure of the host machine.

- [Install the 12c Product Distributions on the Target Machine](#)  
Oracle recommends an Out-of-Place approach for upgrade. Therefore, you must install the 12c product distributions in a new Oracle home on the target machine.
- [Upgrade the Target 64-bit Environment Using the Standard Upgrade Procedure](#)  
After installing the product on the target machine, you must upgrade each product component individually using an Upgrade Utility specified in the component-specific upgrade guide and complete any post-upgrade tasks.

## Procure the Hardware That Supports the Upgrade's 64-bit Software Requirement

Make sure that you have supported target hardware in place before you begin the upgrade process.

## Stop All Processes

Before upgrading, you must stop all processes, including Managed Servers, the Administration Server, and Node Manager, if they are started on the host.

 **Note:**

Ensure that the Database is up and running, during the upgrade.

### 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 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 2:** 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')
```

```
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.

To stop the Administration Server, use the `stopWebLogic` script:

- (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.

### Step 3: Stop Node Manager

To stop Node Manager, close the command shell in which it is running.

Alternatively, after having set the `nodemanager.properties` attribute `QuitEnabled` to `true` (the default is `false`), you can use WLST to connect to Node Manager and shut it down. See `stopNodeManager` in *WLST Command Reference for WebLogic Server*.

## Back Up All Files from the 32-bit Host Machine

Make sure that you have created a complete backup of your entire 11g deployment before you begin the upgrade process. These files can be used if there is an issue during the migration and you have to restart the process.



### Note:

If the upgrade from 32-bit to 64-bit takes place on the same machine, there is a risk of corrupting the source environment if the upgrade fails.

See [Backing Up Your Environment](#) in *Oracle Fusion Middleware Administrator's Guide*.

During the upgrade you must have access to the contents of the following:

- `11g_DOMAIN_HOME`
- `11g/nodemanager` directory located in `11g_ORACLE_HOME/wlserver/common/`

Some of the backup and recovery procedures described in [Backing Up Your Environment](#) in *Oracle Fusion Middleware Administrator's Guide* are product-specific. Do not proceed with the upgrade until you have a complete backup.

## Set Up the Target 64-bit Machine with the 11g Host Name and IP Address

The host name and IP address of the target machine must be made identical to the host. This requires you to change the IP address and name of the source machine or decommission the source machine to avoid conflicts in the network.

The process of changing an IP address and host name vary by operating system. Consult your operating system's administration documentation for more information.

## Restore the 11g Backup from 32-bit Host to 64-bit Host

Restore the files you backed from the 32-bit host using the same directory structure that was used in 11g. The directory structure on the target machine must be identical to the structure of the host machine.

See [Recovering Your Environment](#) in *Oracle Fusion Middleware Administrator's Guide*.



## Install the 12c Product Distributions on the Target Machine

Oracle recommends an Out-of-Place approach for upgrade. Therefore, you must install the 12c product distributions in a new Oracle home on the target machine.

Refer to the component-specific installation guides for the component(s) you are installing.

## Upgrade the Target 64-bit Environment Using the Standard Upgrade Procedure

After installing the product on the target machine, you must upgrade each product component individually using an Upgrade Utility specified in the component-specific upgrade guide and complete any post-upgrade tasks.

If you are upgrading additional components, see the component-specific upgrade guide.

### Note:

The Node Manager upgrade procedure requires access to the original Node Manager files. Use the 11g Node Manager files that you backed up from the 32-bit source machine as part of [Back Up All Files from the 32-bit Host Machine](#).

## 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 12c.

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 12c (12.2.1.3.0).

### Note:

If your database version is no longer supported, you must upgrade to a supported version before starting an upgrade. See *Upgrading and Preparing Your Oracle Databases for 12c* in *Planning an Upgrade of Oracle Fusion Middleware*.

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

At the time this document was published, the certified JDK for 12c (12.2.1.3.0) was 1.8.0\_131.

Refer to the Oracle Fusion Middleware Supported System Configurations information on the Oracle Technical Resources 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`.

## Updating Policy Files when Using Enhanced Encryption (AES 256)

If you plan to use enhanced encryption, such as Advanced Encryption Standard (AES 256), in your upgraded environment, Oracle recommends that you apply the latest required policy files to the JDK before you upgrade.

The Java platform defines a set of APIs spanning major security areas, including cryptography, public key infrastructure, authentication, secure communication, and access control. These APIs allow developers to easily integrate security mechanisms into their application code.

Some of the security algorithms used in Fusion Middleware 12c (12.2.1.3.0) require additional policy files for the JDK. See [Java Cryptography Architecture Oracle Providers Documentation](#).

### Note:

If you attempt to use enhanced encryption without applying these policy files to the JDK before you begin the upgrade, the upgrade can fail and you must restore the entire pre-upgrade environment and start the upgrade from the beginning.

## 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.

For more information, see [Using the Archival and Purge Utilities for Controlling Data Growth](#).

## Creating a Non-SYSDBA User to Run the Upgrade Assistant

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

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.

### Notes:

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. Note that privileges required for running the Upgrade Assistant may change from release to release.

By default, the `v$xsatrans$` table does not exist. You must run the `XAVIEW.SQL` script to create this table before creating the user. Moreover, the `grant select` privilege on the `v$xsatrans$` table is required only by Oracle Identity Governance . If you do not require Oracle Identity Governance for configuration, or if you do not have the `v$xsatrans$` table, then remove the following line from the script:

```
grant select on v$xsatrans$ to FMW with grant option;
```

In the example below, `<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_DATA_FILES to FMW with grant option;
grant select on SYS.V_$ASM_DISKGROUP to FMW with grant option;
grant select on v$xsatrans$ 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 task enables you to review the list of available schemas before you begin the upgrade by querying the schema version registry. The registry contains schema information such as version number, component name and ID, date of creation and modification, and custom prefix.

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 ORDER BY MRC_NAME, COMP_ID;
```

2. Examine the report that is generated.

If an upgrade is not needed for a schema, the `schema_version_registry` table retains the schema at its pre-upgrade version.

3. Note the schema prefix name that was used for your existing schemas. You will use the same prefix when you create new 12c schemas.

### Notes:

- If you used an OID-based policy store in 11g, 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 12c (12.2.1.3.0). Do not attempt to upgrade a domain that includes components that are not yet available for upgrade to 12c (12.2.1.3.0).

# Updating Database Parameters for Oracle Identity Manager

You need to verify and update a few database parameters before upgrading the Oracle Identity Manager to 12c (12.2.1.3.0).

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.

## Updating Connectors for Oracle Identity Manager

Update the existing connectors if they are not supported for Oracle Identity Manager 12c (12.2.1.3.0).

Complete the following steps:

1. Go to the [Oracle Identity Manager Connectors Certification](#).
2. By using the certification information table, verify if the existing connectors are supported for 12c (12.2.1.3.0).
3. Are the existing connectors supported for 12c (12.2.1.3.0)?
  - **Yes:** Skip this procedure and proceed to the next upgrade procedure.
  - **No:** Update the required connectors. See [Oracle Identity Governance 12c Connectors](#).

## 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:

Run this report until no pending issues are listed in the report. 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 11.1.2.3.0 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` along with `ReadMe.doc` at the following location on My Oracle Support: [My Oracle Support document ID 2308933.1](#)

The `ReadMe.doc` contains information about how to generate and analyze the pre-upgrade reports.

## 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
<code>oim.mwhome</code>	Specify the absolute path to the Middleware home of the existing installation. For example: <code>/Oracle/Middleware</code>
<code>oim.oimhome</code>	Specify the absolute path to the existing OIM home. For example: <code>/Oracle/Middleware/Oracle_IDM1</code>
<code>oim.javahome</code>	Specify the absolute path to the Java home. Ensure that you point to JAVA 8.
<code>oim.wlshome</code>	Specify the absolute path to the WebLogic Server home. For example: <code>/Oracle/Middleware/wlserver_10.3</code>
<code>oim.domain</code>	Specify the absolute path to the Oracle Identity Manager domain home. For example: <code>/Oracle/Middleware/user_projects/domains/IAMGovernanceDomain</code>



**Table 2-2 (Cont.) Parameters to be Specified in the preupgrade\_report\_input.properties File**

Parameter	Description
<code>oim.oimhost</code>	Specify the hostname of Oracle Identity Manager. For example: <code>oim.example.com</code>
<code>oim.oimport</code>	Specify the port of the Oracle Identity Manager server. For example: <code>14000</code>
<code>oim.username</code>	Specify the Oracle Identity Manager Admin username. For example: <code>xelsysadm</code>
<code>oim.targetVersion</code>	Specify the target version of the Oracle Identity Manager, that is, <code>12.2.1.3.0</code> .
<code>oim.jdbcurl</code>	Specify the JDBC URL for Oracle Identity Manager in one of the following formats: <code>host:port/service_name</code> or <code>host:port:sid</code>
<code>oim.oimschemaowner</code>	Specify the name of the OIM schema owner.
<code>oim.mdsjdbcurl</code>	Specify the MDS JDBC URL in the one of the following formats: <code>host:port/service_name</code> or <code>host:port:sid</code>
<code>oim.mdsschemaowner</code>	Specify the name of the MDS schema owner.
<code>oim.databaseadminname</code>	Specify the user with DBA privilege. For example, <code>FMW</code> or <code>sys</code> as <code>sysdba</code> .
<code>oim.outputreportfolder</code>	Specify the absolute path to the directory where you want the reports to be generated ( <code>OIM_preupgrade_reports</code> ). Ensure that this directory has read and write permissions.

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.

- **OIM Admin Password:** Enter the password of the Oracle Identity Manager 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
Status of OIM System Property – XL.AllowedBackURLs	This report provides the status of the system property related to setting the back URLs in Oracle Identity Manager.
Changes to SCIM-JWT in 12c	This report lists the new SCIM URLs published during 12c (12.2.1.3.0). You must use the new URLs instead of the old ones.
Potential upgrade issues for User Defined Attributes	This report lists the potential issues with the User Defined Field (UDF) defined in Oracle Identity Manager 11.1.2.3.0 during upgrade.
Status of Mandatory Database Components	This report lists the installation status of the mandatory Database components which are required for upgrade.
Status of Mandatory deletion of OIM Authentication Jar(s)	This report lists the status of a few mandatory jars that need to be deleted before upgrade.
Full MDS Export 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.
OIM-OMSS Integration Pre-Upgrade Report	This report gives the deprecation information about the Oracle Mobile Security Services (OMSS) with Oracle Identity Manager in 12c (12.2.1.3.0).
Status of Domain Configuration	This report lists the applications (if any) that are in stage mode.
Status of Mandatory DB Privilege	This report lists the missing mandatory database privileges that are required for upgrade.
Status of data associated with access policies	In 12c, access policies are associated with application instances instead of resource object. To handle the same, this report lists in-consistent data (if present) in the Oracle Identity Manager 11.1.2.3.0.

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

<b>Report Name</b>	<b>Description and Action Item</b>
Authorization Policy backup of source environment	This report lists the details regarding the Oracle Identity Manager authorization policy backup taken prior to upgrade.
Information about Schedule Jobs against Schedule task named as OIM Data Purge Task on source environment	This report provides important information regarding one of the schedule tasks which will be available after upgrade.
Obsolete templates existence status on source environment	This report lists obsolete templates that are present in the source domain prior to upgrade.
Obsolete applications existence status on source environment	This report lists obsolete applications that are present in the source domain prior to upgrade.
soaOIMLookupDB data source status on source environment	This report lists non-transactional soaOIMLookupDB data sources in the source domain prior to upgrade.
Status of OIM default keystore in KSS on source environment	This report lists the OIM default keystore if it's present in the KSS of the source domain prior to upgrade.

# 3

## Upgrading Oracle Identity Manager Single Node Environments

You can upgrade Oracle Identity Manager from Release 11g Release 2 (11.1.2.3.0) to Oracle Identity Governance 12c (12.2.1.3.0) .

 **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.
- [Installing Product Distributions](#)  
Before beginning your upgrade, download Oracle Fusion Middleware Infrastructure, Oracle SOA Suite, and Oracle Identity Manager 12c (12.2.1.3.0) distributions on the target system and install them using Oracle Universal Installer.
- [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.
- [Creating the Required 12c Schemas Using RCU \(Optional\)](#)  
When upgrading from 11g, you must create the required 12c schemas. If your setup has non-SSL ports open, you can use the Upgrade Assistant to create schemas by using the default schema settings. In case of SSL enabled setup, you can use the Repository Creation Utility (RCU) to create customized schemas. This procedure describes how to create schemas using the RCU. Information about using the Upgrade Assistant to create schemas is covered in the upgrade procedures.
- [Tuning Database Parameters for Oracle Identity Manager](#)  
Before you upgrade the schemas, you must tune the Database parameters for Oracle Identity Manager.

- [Stopping Servers and Processes](#)  
Before you run the Upgrade Assistant to upgrade your 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.
- [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.
- [About Reconfiguring the Domain](#)  
Run the Reconfiguration Wizard to reconfigure your domain component configurations to 12c (12.2.1.3.0).
- [Upgrading Domain Component Configurations](#)  
After reconfiguring the domain, use the Upgrade Assistant to upgrade the domain *component* configurations inside the domain to match the updated domain configuration.
- [Post-Upgrade Task](#)  
After upgrading from 11g to 12c, you need to copy any custom configuration present in your 11g Middleware home to the 12c Oracle home.
- [Copying folders to the 12c Oracle Home](#)  
When upgrading to 12c, you must manually copy some folders to the 12c Oracle Home, if those folders are having file system dependent data.
- [Starting the Servers](#)  
After you upgrade Oracle Identity Manager, start the servers.
- [Upgrading Oracle Identity Manager Design Console](#)  
Upgrade the Oracle Identity Manager Design Console after you upgrade the Oracle Identity Manager (OIM) domain component configurations.
- [Completing the Post-Upgrade Tasks for SSL Enabled Setup](#)  
If you are upgrading an Oracle Identity Manager SSL enabled setup, you must perform the required post-upgrade tasks to complete the upgrade process.
- [Installing Standalone Oracle BI Publisher](#)  
When you upgrade Oracle Identity Manager 11.1.2.3.0 to Oracle Identity Governance 12c (12.2.1.3.0), the embedded Oracle BI Publisher present in the 11.1.2.3.0 deployment, is removed. Therefore, you must install a new standalone Oracle BI Publisher 12c (12.2.1.3.0) post upgrade, for configuring the Oracle Identity Governance reports.
- [Tuning Application Module for User Interface](#)  
After you successfully upgrade the Oracle Identity Manager middle-tier, tune the Application Module (AM).

## 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
<p><b>Required</b> If you have not done so already, review the introductory topics in this guide and complete the required pre-upgrade tasks.</p>	<p>See:</p> <ul style="list-style-type: none"> <li>• <a href="#">Introduction to Upgrading Oracle Identity Manager to 12c (12.2.1.3.0)</a></li> <li>• <a href="#">Pre-Upgrade Requirements</a></li> </ul>
<p><b>Required</b> Complete the necessary pre-upgrade tasks specific to Oracle Identity Manager.</p>	<p>See <a href="#">Completing the Pre-Upgrade Tasks for Oracle Identity Manager</a>.</p>
<p><b>Required</b> Install Fusion Middleware Infrastructure 12c (12.2.1.3.0), Oracle SOA Suite12c (12.2.1.3.0) and Oracle Identity Manager12c (12.2.1.3.0) in a new Oracle home.</p>	<p>Install the following products in a <i>new</i> Oracle home on the same host as the 11g production deployment before you begin the upgrade.</p> <ul style="list-style-type: none"> <li>• Fusion Middleware Infrastructure 12c (12.2.1.3.0)</li> <li>• Oracle SOA Suite12c (12.2.1.3.0)</li> <li>• Oracle Identity Manager12c (12.2.1.3.0)</li> </ul> <p>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 Manager 12c (12.2.1.3.0) in one go. See <i>Installing Oracle Identity Governance Using Quick Installer</i> in the <i>Installing and Configuring Oracle Identity and Access Management</i>.</p> <p>The other option is to install these products separately using their respective installers. See <a href="#">Installing Product Distributions</a>.</p>
<p><b>Optional</b> Run a pre-upgrade readiness check.</p>	<p>See <a href="#">Running a Pre-Upgrade Readiness Check</a>.</p>
<p><b>Optional</b> Start the Repository Creation Utility (RCU) to create the required 12c database schemas.  This step is not required for non-SSL setup, as the Upgrade Assistant creates the necessary 12c schemas during the upgrade process.  For SSL enabled setup, you must run the RCU to create the necessary 12c schemas.</p>	<p>The schemas you create will vary depending on your existing schema configuration. See <a href="#">Creating the Required 12c Schemas with the RCU</a>.</p>
<p><b>Required</b> Tune the Database parameters for Oracle Identity Manager.</p>	<p>See <a href="#">Tuning Database Parameters for Oracle Identity Manager</a> .</p>
<p><b>Required</b> Shut down the 11g servers. This includes the Administration Server, Managed Servers, Node Manager, and system components.  Ensure that the Database is up during the upgrade.</p>	<p><b>WARNING:</b> Failure to shut down your servers during an upgrade may lead to data corruption. See <a href="#">Stopping Servers and Processes</a>.</p>

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

Task	Description
<p><b>Required</b></p> <p>Start the Upgrade Assistant to upgrade the 11g database schemas and to migrate all active (in flight) instance data.</p>	<p>See <a href="#">Upgrading Product Schemas</a>.</p> <p><b>NOTE:</b> The upgrade of active instance data is started automatically when running the Upgrade Assistant. Once the data is successfully upgraded to the new 12c (12.2.1.3.0) environment, you can close the Upgrade Assistant. The closed instances will continue to upgrade through a background process.</p>
<p><b>Required</b></p> <p>Tune the application module for Oracle Identity Manager.</p>	<p>See <a href="#">Tuning Application Module for User Interface</a>.</p>
<p><b>Required</b></p> <p>Start the Reconfiguration Wizard to reconfigure the domain.</p>	<p>During an upgrade, the Configuration Wizard is run in reconfiguration mode to update the existing domain to use the newly installed software.</p> <p>See <a href="#">Reconfiguring the Domain Using the Reconfiguration Wizard</a>.</p>
<p><b>Required</b></p> <p>Start the Upgrade Assistant (again) to upgrade Oracle Identity Manager domain component configurations.</p>	<p>The Upgrade Assistant is used to update the reconfigured domain's component configurations.</p> <p>See <a href="#">Upgrading Domain Component Configurations</a>.</p>
<p><b>Required</b></p> <p>Start the servers.</p>	<p>See <a href="#">Starting the Servers</a>.</p>
<p><b>Required</b></p> <p>Upgrade the Oracle Identity Manager Design Console to 12c (12.2.1.3.0).</p>	<p>See <a href="#">Upgrading Oracle Identity Manager Design Console</a>.</p>
<p><b>Optional</b></p> <p>Perform the post-upgrade tasks for SSL enabled setup.</p>	<p>See <a href="#">Completing the Post-Upgrade Tasks for SSL Enabled Setup</a>.</p>
<p><b>Optional</b></p> <p>When you upgrade to Oracle Identity Governance 12c (12.2.1.3.0), the embedded Oracle BI Publisher present in the 11.1.2.3.0 deployment is removed. Therefore, you must install a new standalone Oracle BI Publisher 12c (12.2.1.3.0) post upgrade, and integrate it with Oracle Identity Governance 12c (12.2.1.3.0) to configure the Oracle Identity Governance reports.</p>	<p>See, <a href="#">Installing Standalone Oracle BI Publisher</a>.</p>

## Completing the Pre-Upgrade Tasks for Oracle Identity Manager

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

- [Updating Server Wallets to Remove MD5 Algorithm](#)

If the existing keystore has a certificate which is invalid with the JDK that is used to install 12c (12.2.1.3.0) binaries, you must generate the keystore and place it in the `DOMAIN_HOME/config/fmwconfig` directory.

- [Updating DB Wallets to Remove MD5 Algorithm \(For SSL Enabled Setup\)](#)  
If you have SSL enabled setup, update all of the DB wallets to remove any MD5 algorithms, as 12c (12.2.1.3.0) uses JDK 8 which does not support MD5 algorithm.
- [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 Servers and Database before you proceed with the Oracle Identity Manager upgrade.

## Updating Server Wallets to Remove MD5 Algorithm

If the existing keystore has a certificate which is invalid with the JDK that is used to install 12c (12.2.1.3.0) binaries, you must generate the keystore and place it in the `DOMAIN_HOME/config/fmwconfig` directory.

If the default keystore has MD5 algorithm, then the upgrade readiness check and the examine phase of OIM configuration upgrade will fail.

To verify the validity of the certificate, do the following:

1. Check for the `jdk.certpath.disabledAlgorithms` property in the `JAVA_HOME/jre/lib/security/java.security` file.

For example:

```
jdk.certpath.disabledAlgorithms=MD2, MD5, RSA keySize < 1024
```

2. Check for the certificate algorithm in the existing keystore by doing the following:
  - a. For default keystore, `DOMAIN_HOME/config/fmwconfig/default-keystore.jks`, run the following command from the `JAVA_HOME/jre/bin` directory:

```
./keytool -list -v -keystore DOMAIN_HOME/config/fmwconfig/default-keystore.jks
```

If you are using the custom keystores, that is, `DOMAIN_HOME/config/fmwconfig/name_of_custom_store`, run the following command from the `JAVA_HOME/jre/bin` directory:

```
./keytool -list -v -keystore DOMAIN_HOME/config/fmwconfig/custom_keystore.jks
```

This command displays the keystore data. Enter the keystore password when prompted.

- b. Check for the `Signature algorithm name` field value in the output of the above command. If the value of `Signature algorithm name` field and the `jdk.certpath.disabledAlgorithms` property has MD5 algorithm, then the given keystore will not be valid after upgrade.



If the keystore is not valid after upgrade, the following error is seen in the server logs while executing the request use cases after upgrade, and none of the request use cases will be successful:

```
Caused by: java.security.cert.CertPathValidatorException:
Algorithm
constraints check failed: MD5withRSA
```

3. Generate a certificate signing request from the temporary keystore and load it into `<cert_req>.csr` with the alias `xell` using the `keytool` command:

```
keytool -certreq -alias xell -file /tmp/<cert_req>.csr -
keypass keypassword -keystore /temporary location/default-keystore.jks -
storepass storepassword -storetype jks
```

 **Note:**

The following commands can be used to generate both default keystore and custom keystores.

- `./keytool -genkeypair -keystore new_keystore_location/new_keystore_name.jks -keyalg supported_algorithm_name -sigalg SHA256withRSA -validity validity_period -keypass key_password -storepass keystore_password -alias xell -dname vaild_name -keysize key_size`

For example:

```
./keytool -genkeypair -keystore /scratch/default-keystore.jks
-keyalg RSA -sigalg SHA256withRSA -validity 3600 -
keypass password -storepass password -alias xell -dname
"CN=oimhost,OU=Identity,O=ABC,C=XX" -keysize 2048
```

- `./keytool -exportcert -keystore new_keystore_location/new_keystore_name.jks -rfc -file new_keystore_location/certificate_name.cer`

For example:

```
./keytool -exportcert -keystore /scratch/default-keystore.jks -v
-alias xell -storepass password -rfc -file /scratch/cert.cer
```

- `./keytool -importcert -keystore new_keystore_location/new_keystore_name.jks -alias xeltrusted -file new_keystore_location/certificate_name.cer -storepass keystore_password`

Click **Yes** to confirm the action.

For example:

```
./keytool -importcert -keystore /scratch/default-keystore.jks -
alias xeltrusted -file /scratch/cert.cer -storepass password
```

4. Import the newly generated keystore into the existing keystore `DOMAIN_HOME/config/fmwconfig/default-keystore.jks` by running the following command:

```
./keytool -importkeystore -srckeystore new_keystore_location/
new_keystore_name.jks -destkeystore DOMAIN_HOME/config/fmwconfig/
default-keystore.jks -srcstorepass source_keystore_password -
deststorepass destination_keystore_password -noprompt
```

For example:

```
./keytool -importkeystore -srckeystore /scratch/default-
keystore.jks -destkeystore domain_home/config/fmwconfig/default-
keystore.jks -srcstorepass password -deststorepass password -
noprompt
```

5. Log in to Enterprise Manager console and update the `xell` named CSF key under `oim` map, with the password value which is used above to generate the new key in keystore. In the above example, the password used is `password`.
6. Move the `<export file>.cert` and the `<cert_req>.csr` to the `DOMAIN_HOME/config/fmwconfig/` location.

```
cp /tm/xell.csr DOMAIN_HOME/config/fmwconfig/
cp /tmp/xlserver.cert DOMAIN_HOME/config/fmwconfig/
```

#### Note:

For more information about using the `keytool` command, see [keytool](#) in the *Java Platform, Standard Edition Tools Reference*.

#### Note:

The procedure described in this section for regenerating the `default-keystore.jks` or custom keystore includes self-signed certificates. If CA signed certificate is required, follow the standard process for the same, that is, generate the CSR and import the signed certificates in the keystore.

During bootstrap process in OIM, the `default-keystore.jks` keystore will be configured in Keystore Service (KSS) out-of-the-box. In case of custom keystore, upload the given custom keystore to KSS after completing the upgrade. After you upload the given custom keystore to KSS, restart the servers.

For more information about the Keystore Service commands, see *OPSS Keystore Service Commands in WLST Command Reference for Infrastructure Security*.

## Updating DB Wallets to Remove MD5 Algorithm (For SSL Enabled Setup)

If you have SSL enabled setup, update all of the DB wallets to remove any MD5 algorithms, as 12c (12.2.1.3.0) uses JDK 8 which does not support MD5 algorithm.

### Note:

All these steps in this procedure must be performed on the Database server. That is, on the server where OIM database is installed.

To update the DB wallet, do the following:

1. Create an Oracle Wallet with default trusted certificate using the following command:

```
./orapki wallet create -wallet <trust_wallet_name> -pwd password
```

For example:

```
./orapki wallet create -wallet trust_wallet.p12 -pwd password
```

2. Add a self-signed certificate in the wallet with the distinguished name (DN) as CN=root\_test,C=US using the following command:

```
./orapki wallet add -wallet trust_wallet_name -dn 'dn_name' -keysize 2048 -sign_alg sha256 -self_signed -validity 3650 -pwd password_of_wallet
```

For example:

```
./orapki wallet add -wallet trust_wallet.p12 -dn 'CN=root_test,C=US' -keysize 2048 -sign_alg sha256 -self_signed -validity 3650 -pwd password
```

3. Export the self-signed trust certificate from the Oracle wallet to use it to sign other certificates, using the following command:

```
./orapki wallet export -wallet trust_wallet_name -dn 'dn_name' -cert trust_cert_file_name -pwd password_of_wallet
```

For example:

```
./orapki wallet export -wallet trust_wallet.p12 -dn 'CN=root_test,C=US' -cert wallet_trusted.cert -pwd password
```

4. You already have an Oracle Wallet with User Certificate identified. The user wallet is, *DB\_HOME/bin/user\_wallet.p12*. The DN of this user certificate is CN=Customer,OU=Customer,O=Customer,L=City,ST=NY,C=US. Remove the existing user certificate from this wallet using the following command:

Where, *DB\_HOME* is the server where OIM database is installed.

```
./orapki wallet remove -wallet user_wallet_name -pwd password_of_existing_wallet -dn 'DN_name' -user_cert
```

For example:

```
./orapki wallet remove -wallet user_wallet.p12 -pwd password -dn 'CN=Customer,OU=Customer,O=Customer,L=City,ST=NY,C=US' -user_cert
```

5. You already have an Oracle Wallet with Requested Certificate identified. The user wallet is, `DB_HOME/bin/user_wallet.p12`. The DN of this requested certificate is `CN=Customer,OU=Customer,O=Customer,L=City,ST=NY,C=US`. Remove the existing requested certificate from this wallet using the following command:

```
./orapki wallet remove -wallet user_wallet_name -dn 'DN_name' -
cert_req -pwd password_of_existing_wallet
```

For example:

```
./orapki wallet remove -wallet user_wallet.p12 -dn
'CN=Customer,OU=Customer,O=Customer,L=City,ST=NY,C=US' -cert_req -pwd
password
```

6. You already have an Oracle Wallet with Trust Certificate identified. The user wallet is, `DB_HOME/bin/user_wallet.p12`. The DN of this trust certificate is `CN=root_test,C=US`. Remove the existing trust certificate from this wallet using the following command:

```
./orapki wallet remove -wallet user_wallet_name -pwd password-of-
existing_wallet -dn 'DN_name' -trusted_cert
```

For example:

```
./orapki wallet remove {wallet user_wallet.p12 -pwd password -dn '
CN=root_test,C=US' -trusted_cert
```

7. Add a user certificate in the existing user wallet with a distinguished name as `CN=Customer,OU=Customer,O=Customer,L=City,ST=NY,C=US` using the following command:

```
./orapki wallet add -wallet user_wallet_name -dn 'dn_name' -keysize
2048 -sign_alg sha256 -pwd password_of_existing_wallet
```

For example:

```
./orapki wallet add -wallet user_wallet.p12 -dn
'CN=Customer,OU=Customer,O=Customer,L=City,ST=NY,C=US' -keysize 2048 -
sign_alg sha256 -pwd password
```

8. Export the user certificate request to a file using the following command:

```
./orapki wallet export -wallet user_wallet_name -dn 'dn_name' -request
CSR_file_name -pwd password_of_existing_wallet
```

For example:

```
./orapki wallet export -wallet user_wallet.p12 -dn
'CN=Customer,OU=Customer,O=Customer,L=City,ST=NY,C=US' -request
server_creq.csr -pwd password
```

9. Sign the user certificate request using the trusted wallet that was created above, using the following command:

```
./orapki cert create -wallet trusted_wallet_name-request
CSR_file_name -cert user_cert_file_name sign_alg sha256 -pwd
password_of_exiting_user_wallet
```

For example:

```
./orapki cert create -wallet trust_wallet.p12 -request server_creq.csr
-cert wallet_user.cert -sign_alg sha256 - validity 3650 -pwd
password
```

10. Add the trusted certificate `wallet_trusted.cert` that you created using the above procedure to the wallet, by running the following command:

```
./orapki wallet add -wallet user_wallet_name -trusted_cert -cert
trust_cert_file_name -pwd password_of_existing_user_wallet
```

For example:

```
./orapki wallet add -wallet user_wallet.p12 -trusted_cert -cert
wallet_trusted.cert -pwd password
```

11. Add the signed user certificate to the Oracle wallet using the following command:

```
./orapki wallet add -wallet user_wallet -user_cert -cert
user_cert_file_name -pwd password_of_existing_user_wallet

./orapki wallet add -wallet user_wallet.p12 -user_cert -cert
wallet_user.cert -pwd password
```

12. Remove the DB trusted certificate from server keystore. In case of demo identity and demo trust, remove from default-keystore.jks, and in case of custom identity and custom trust, remove it from the custom trust keystore, using the following command:

```
./keytool -delete -alias alias_of_db_cert -keystore custom_trust_store
-storepass password-of-existing-trust-keystore
```

For example:

```
./keytool -delete -alias dbtrusted -keystore DOMAIN_HOME/config/
fmwconfig/custom_trust_store.jks -storepass password
```

13. Import self signed DB certificate in trust wallet using the following command:

```
keytool -import -trustcacerts -alias <alias_of_db_cert> -noprompt
-keystore custom_trust_store -file DB_Trust_cert_file -storepass
password_of_existing_trust_keystore
```

For example:

```
keytool -import -trustcacerts -alias dbtrusted -noprompt -
keystore DOMAIN_HOME/config/fmwconfig/custom_trust_store.jks -file /
DB_HOME/bin/wallet_trusted.cert -storepass password
```

## 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` 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. Run the following command to increase the limit of open files, so that you do not run into memory issues:

```
limit maxproc 16384
```

To ensure that the settings persist after a reboot, add the `limit maxproc 16384` entry to `/etc/security/limits.conf` or `/etc/security/limits.d` file.

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 Servers and Database before you proceed with the Oracle Identity Manager upgrade.

To enable non-SSL ports for servers, complete the following steps:

1. Log in to the WebLogic Server Administration Console.
2. Click **Environment** > **Servers** > and select the required admin server.
3. On the **Settings for Server** page, click the **Configuration** tab, and then click **General**.
4. Click **Lock & Edit**.
5. Select **Listen Port Enabled**. The default port is 14000.
6. Repeat the [step 1](#) through [step 5](#) for each required server in the domain.

 **Note:**

After you complete the upgrade, you can undo these changes as required.

For database: Ensure that database listener is listening on the same TCP port for database servers that you provided to upgrade assistant as parameters. For more information, see [Enabling SSL for Oracle Identity Governance DB](#).

## Installing Product Distributions

Before beginning your upgrade, download Oracle Fusion Middleware Infrastructure, Oracle SOA Suite, and Oracle Identity Manager 12c (12.2.1.3.0) distributions on the target system and install them using Oracle Universal Installer.

 **Note:**

The 12c binaries are installed in a different location from the previous 11g binaries. You can install 12c 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 quick installer (`fmw_12.2.1.3.0_idmquickstart_generic.jar`). The quick installer installs the Infrastructure, Oracle SOA Suite, and Oracle Identity Manager 12c (12.2.1.3.0) 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 Quick 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 12c (12.2.1.3.0) separately. To do this, complete the following steps:

1. Sign in to the target system.
2. Download the following from [Oracle Technical Resources](#) or [Oracle Software Delivery Cloud](#) to your target system:
  - Oracle Fusion Middleware Infrastructure (fmw\_12.2.1.3.0\_infrastructure\_generic.jar)
  - Oracle SOA Suite (fmw\_12.2.1.3.0\_soa\_generic.jar)
  - Oracle Identity Manager (fmw\_12.2.1.3.0\_idm\_generic.jar)
3. Change to the directory where you downloaded the 12c (12.2.1.3.0) product distribution.
4. Start the installation program for Oracle Fusion Middleware Infrastructure:
  - (UNIX) `JAVA_HOME/bin/java -jar fmw_12.2.1.3.0_infrastructure_generic.jar`
  - (Windows) `JAVA_HOME\bin\java -jar fmw_12.2.1.3.0_infrastructure_generic.jar`
5. 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.

6. On the Welcome screen, review the information to make sure that you have met all the prerequisites. Click **Next**.
7. 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**.

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

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

9. On the Installation Type screen, select the following:
  - For Infrastructure, select **Fusion Middleware Infrastructure**
  - For Oracle SOA Suite, select **Oracle SOA Suite**
  - For Oracle Identity Manager, select **Oracle Identity and Access Management**

Click **Next**.

10. 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).

11. 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.

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

13. 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.

14. 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 12c (12.2.1.3.0), run the following installer:

- (UNIX) `JAVA_HOME/bin/java -jar fmw_12.2.1.3.0_soa_generic.jar`
- (Windows) `JAVA_HOME\bin\java -jar fmw_12.2.1.3.0_soa_generic.jar`

For installing Oracle Identity Manager 12c (12.2.1.3.0), run the following installer:

- (UNIX) `JAVA_HOME/bin/java -jar fmw_12.2.1.3.0_idm_generic.jar`



- (Windows) `JAVA_HOME\bin\java -jar fmw_12.2.1.3.0_idm_generic.jar`

 **Note:**

By using the `opatch` tool, apply the latest recommended patchsets from Oracle Support. Complete only the binary installation of patchsets and follow any post-patch steps after the upgrade process is complete. This provides the latest known fixes for upgrade process, if any.

For more information about installing Oracle Identity Manager 12c (12.2.1.3.0), 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 12c 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 <b>Note:</b> 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 <i>silent mode</i> (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 <i>attribute</i>	Optional	Sets the logging level, specifying one of the following attributes: <ul style="list-style-type: none"> <li>• TRACE</li> <li>• NOTIFICATION</li> <li>• WARNING</li> <li>• ERROR</li> <li>• INCIDENT_ERROR</li> </ul> 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
<code>-logDir location</code>	Optional	<p>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.</p> <p>The default locations are:</p> <p>(UNIX)</p> <pre>ORACLE_HOME/ oracle_common/upgrade/ logs ORACLE_HOME/ oracle_common/upgrade/ temp</pre> <p>(Windows)</p> <pre>ORACLE_HOME\oracle_commo n\upgrade\logs ORACLE_HOME\oracle_commo n\upgrade\temp</pre>
<code>-help</code>	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.

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 11.1.2.3.0 Administrator server domain directory.
- Specify schema credentials to connect to the selected schema: **Database Type**, **DBA User Name**, and **DBA Password**. 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**.

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**.

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 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 <i>ORACLE_HOME</i> /oracle_common/ upgrade/logs	The directory location of the generated log file.	No action required.
Readiness report location <i>ORACLE_HOME</i> /oracle_common/ upgrade/logs	The directory location of the generated readiness report.	No action required.

**Table 3-3 (Cont.) Readiness Report Elements**

Report Information	Description	Required Action
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 Tue May 30 11:15:52 EDT 2016
Log file is located at: NEW_ORACLE_HOME/oracle_common/upgrade/logs/
ua2016-05-30-11-14-06AM.log
Readiness Check Report File: NEW_ORACLE_HOME/oracle_common/upgrade/logs/
readiness2016-05-30-11-15-52AM.txt
```

```
Starting readiness check of components.
```

```
Oracle Metadata Services
```

```
Starting readiness check of Oracle Metadata Services.
```

```
Schema User Name: DEV11_MDS
```

```
Database Type: Oracle Database
```

```
Database Connect String: machinename@yourcompany.com
```

```
VERSION Schema DEV11_MDS is currently at version 12.1.1.1.0.
```

```
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
EXCEPTION Schema is missing a required procedure:
GETREPOSITORYFEATURES
Completed schema test: TEST_REQUIRED_PROCEDURES --> Test that the
schema contains all the required stored procedures +++ FAIL
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
Completed index test for table MDS_ATTRIBUTES: TEST_REQUIRED_INDEXES
--> Test that the table contains all the required indexes +++ PASS
Starting index test for table MDS_COMPONENTS: TEST_REQUIRED_INDEXES
--> Test that the table contains all the required indexes
Completed index test for table MDS_TXN_LOCKS: TEST_REQUIRED_INDEXES
--> Test that the table contains all the required indexes +++ PASS
Starting schema test: TEST_REQUIRED_TRIGGERS Test that the schema
has all the required triggers
Completed schema test: TEST_REQUIRED_TRIGGERS --> Test that the
schema has all the required triggers +++ 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_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_UNEXPECTED_PROCEDURES Test that the
schema does not contain any unexpected stored procedures
Completed schema test: TEST_UNEXPECTED_PROCEDURES --> Test that the
schema does not contain any unexpected stored procedures +++ PASS
Starting schema test: TEST_UNEXPECTED_VIEWS Test that the schema
does not contain any unexpected views
Completed schema test: TEST_UNEXPECTED_VIEWS --> Test that the
schema does not contain any unexpected views +++ PASS
Starting index test for table MDS_ATTRIBUTES:
TEST_UNEXPECTED_INDEXES --> Test that the table does not contain any
unexpected indexes
Completed index test for table MDS_ATTRIBUTES:
TEST_UNEXPECTED_INDEXES --> Test that the table does not contain any
unexpected indexes +++ PASS
Completed index test for table MDS_LABELS: TEST_UNEXPECTED_INDEXES
--> Test that the table does not contain any unexpected indexes +++ PASS
Starting index test for table MDS_LARGE_ATTRIBUTES:
TEST_UNEXPECTED_INDEXES --> Test that the table does not contain any
unexpected indexes
Starting schema test: TEST_UNEXPECTED_TRIGGERS Test that the
schema does not contain any unexpected triggers
Completed schema test: TEST_UNEXPECTED_TRIGGERS --> Test that the
schema does not contain any unexpected triggers +++ 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 MDS_ATTRIBUTES:
TEST_COLUMN_DATATYPES_V2 --> Test that all table columns have the
proper datatypes
Completed datatype test for table MDS_ATTRIBUTES:
TEST_COLUMN_DATATYPES_V2 --> Test that all table columns have the
proper datatypes +++ PASS
Starting datatype test for table MDS_COMPONENTS:
TEST_COLUMN_DATATYPES_V2 --> Test that all table columns have the
proper datatypes
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_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 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 11.2.0.3.0 - 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:
FAILURE.
```

If you are running the 12.1.3.0 version of Oracle Fusion Middleware IAU Schemas, and those schemas were upgraded from 11g (11.1.1.7 and later) or 12c (12.1.2.0), your readiness check may fail with the following error:



**Note:**

This is not applicable for Oracle Identity Manager.

```
Starting index test for table IAU_COMMON: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes
INFO Audit schema index DYN_EVENT_CATEGORY_INDEX in table IAU_COMMON is
```

missing the required columns or index itself is missing. This maybe caused by a known issue, anyway, this missing index will be added in 12.2.2 upgrade.

INFO Audit schema index DYN\_EVENT\_TYPE\_INDEX in table IAU\_COMMON is missing the required columns or index itself is missing. This maybe caused by a known issue, anyway, this missing index will be added in 12.2.2 upgrade.

INFO Audit schema index DYN\_TENANT\_INDEX in table IAU\_COMMON is missing the required columns or index itself is missing. This maybe caused by a known issue, anyway, this missing index will be added in 12.2.2 upgrade.

INFO Audit schema index DYN\_USER\_INDEX in table IAU\_COMMON is missing the required columns or index itself is missing. This maybe caused by a known issue, anyway, this missing index will be added in 12.2.2 upgrade.

INFO Audit schema index DYN\_COMPONENT\_TYPE\_INDEX in table IAU\_COMMON is missing the required columns or index itself is missing. This maybe caused by a known issue, anyway, this missing index will be added in 12.2.2 upgrade.

INFO Audit schema index DYN\_USER\_TENANT\_INDEX in table IAU\_COMMON is missing the required columns or index itself is missing. This maybe caused by a known issue, anyway, this missing index will be added in 12.2.2 upgrade.


Completed index test for table IAU\_COMMON: TEST\_REQUIRED\_INDEXXES --> Test that the table contains all the required indexes +++ FAIL

 **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 12c using the RCU.

## Creating the Required 12c Schemas Using RCU (Optional)

When upgrading from 11g, you must create the required 12c schemas. If your setup has non-SSL ports open, you can use the Upgrade Assistant to create schemas by using the default schema settings. In case of SSL enabled setup, you can use the Repository Creation Utility (RCU) to create customized schemas. This procedure describes how to create schemas using the RCU. Information about using the Upgrade Assistant to create schemas is covered in the upgrade procedures.

 **Note:**

This step is not required for non-SSL setup, as the Upgrade Assistant creates the necessary 12c schemas during the upgrade process.

For SSL enabled setup, you must run the RCU to create the necessary 12c schemas.

 **Note:**

If you are upgrading from a previous 12c release of Oracle Fusion Middleware, you do not need to re-create these schemas if they already exist. Refer to the steps below to identify the existing schemas in your domain.

The following schemas must exist before you upgrade to 12c. If you are upgrading from 11g, and you are not sure which schemas you currently have, refer to the steps below to identify the existing schemas in your domain. You do not need to re-create these schemas if they already exist.

- **Service Table** schema (*prefix\_STB*). This schema is new in 12c and is required for domain-based upgrades. It stores basic schema configuration information (for example, schema prefixes and passwords) that can be accessed and used by other Oracle Fusion Middleware components during the domain creation. This schema is automatically created when you run the Repository Creation Utility (RCU), where you specify the existing schema owner prefix that you used for your other 11g schemas.

 **Note:**

If the Service Table schema does not exist, you may encounter the error message UPGAST-00328 : The schema version registry table does not exist on this database. If that happens it is necessary to create the service table schema in order to run Upgrade Assistant

- **Oracle Platform Security Services (OPSS)** schema (*prefix\_OPSS*). This schema is required if you are using an OID-based security store in 11g. This schema is automatically created when you run the Repository Creation Utility (RCU). The only supported LDAP-based OPSS security store is Oracle Internet Directory (OID). An LDAP-based policy store is typically used in production environments. You do not need to reassociate an OID-based security store before upgrade. While the Upgrade Assistant is running, you can select the OPSS schema. The Upgrade Assistant upgrades the OID-based security store automatically.

 **Note:**

The 12c OPSS database schema is required so that you can reference the 12c schema during the reconfiguration of the domain. Your domain continues to use the OID-based security store after the upgrade is complete.

To create the 12c schemas with the RCU:

1. (Optional) If you are upgrading from 11g, and you wish to confirm the schemas which are present in your existing domain, then connect to the database as a user with DBA privileges, and run the following code 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 ORDER BY MRC_NAME, COMP_ID ;
```

2. Verify that a certified JDK already exists on your system by running `java -version` from the command line. For 12c (12.2.1.3.0), the certified JDK is 1.8.0\_131 and later.

Ensure that the `JAVA_HOME` environment variable is set to the location of the certified JDK. For example:

- (UNIX) `setenv JAVA_HOME=/home/Oracle/Java/jdk1.8.0_131`
- (Windows) `set JAVA_HOME=C:\home\Oracle\Java\jdk1.8.0_131`

Add `%JAVA_HOME%\bin` to `%PATH%`.

3. Go to the `oracle_common/bin` directory:
  - (UNIX) `NEW_ORACLE_HOME/oracle_common/bin`
  - (Windows) `NEW_ORACLE_HOME\oracle_common\bin`
4. Start the RCU:
  - (UNIX) `./rcu`
  - (Windows) `rcu.bat`
5. On the Welcome screen, click **Next**.
6. On the Create Repository screen, select **Create Repository** and then select **System Load and Product Load**.

If you do not have DBA privileges, select **Prepare Scripts for System Load**. This will generate a SQL script containing all the same SQL statements and blocks that would have been called if the RCU were to execute the actions for the selected components. After the script is generated, a user with the necessary SYS or SYSDBA privileges can execute the script to complete the system load phase. Click **Next**.

7. On the Database Connection Details screen, select the **Database Type** and enter the connection information for the database that hosts the 11g schemas. See the pertinent table below.

**Table 3-4 Connection Credentials for Oracle Databases and Oracle Databases with Edition-Based Redefinition**

Option	Description and Example
Host Name	Specify the name of the server where your database is running in the following format: <code>examplehost.exampledomain.com</code> For Oracle RAC databases, specify the SCAN name or one of the node names in this field.
Port	Specify the port number for your database. The default port number for Oracle databases is 1521.
Service Name	Specify the service name for the database. Typically, the service name is the same as the global database name. For Oracle RAC databases, specify the service name of one of the nodes in this field. For example: <code>examplehost.exampledomain.com</code>
Username	Enter the user name for your database. The default user name is SYS.

**Table 3-4 (Cont.) Connection Credentials for Oracle Databases and Oracle Databases with Edition-Based Redefinition**

Option	Description and Example
Password	Enter the password for your database user.
Role	Select the database user's role from the drop-down list: <b>Normal</b> or <b>SYSDBA</b>

8. On the Select Components screen, select **Select existing prefix** and select the prefix that was used to create the existing 11g schemas from the drop-down menu (for example, `DEV11G`). This prefix is used to logically group schemas together for use in this domain. Select the following schemas:

- If you are upgrading an SSL enabled setup, select the following schemas:
  - User Messaging Service (`prefix_UMS`)
  - Weblogic Services (`prefix_WLS`)
  - Audit services (`prefix_IAU_APPEND` and `prefix_IAU_VIEWER`)

 **Note:**

The Common Infrastructure Services (`prefix_STB`) is selected by default. IAU is greyed out if 11g is configured for Audit Data Store).

- If you are upgrading a non-SSL enabled setup, select the following schemas:
  - Weblogic Services (`prefix_WLS`)
  - Audit services (`prefix_IAU_APPEND` and `prefix_IAU_VIEWER`)

 **Note:**

The Common Infrastructure Services (`prefix_STB`) is selected by default. IAU is greyed out if 11g is configured for Audit Data Store).

 **Note:**

The Common Infrastructure Services (`prefix_STB`) and Oracle Platform Security Services (`prefix_OPSS`) schemas are selected by default if they have not yet been created.

Make a note of the prefix and schema names for the components you are installing as you will need this information when you configure the installation. Click **Next**.

9. In the Checking Prerequisites dialog, verify that the prerequisites check is successful, then click **OK**.

10. On the Schema Passwords screen, specify the passwords for your schema owners.

Make a note of the passwords you enter on this screen as you will need this information while configuring your product installation.

11. On the Map Tablespaces screen, configure the required tablespace mapping for the schemas you want to create.

Click **Next**, then click **OK** in the confirmation dialog. When the progress dialog shows the tablespace creation is complete, click **OK**.

You see the **Encrypt Tablespace** check box only if you have enabled Transparent Data Encryption (TDE) in the database (Oracle or Oracle EBR) when you start the RCU. Select the **Encrypt Tablespace** check box on the Map Tablespaces screen to encrypt all new tablespaces that the RCU creates.

12. Verify the information on the Summary screen and click **Create** to begin schema creation.

This screen contains information about the log files that were created from this RCU operation. Click on the name of a particular log file to view the contents of that file.

13. Review the information on the Completion Summary screen to verify that the operation is completed successfully. Click **Close** to complete the schema creation.

## Tuning Database Parameters for Oracle Identity Manager

Before you upgrade the schemas, you must tune the Database parameters for Oracle Identity Manager.

See [Performance Tuning Guidelines and Diagnostics Collection for Oracle Identity Manager \(OIM\) \(Doc ID 1539554.1\)](#).

## Stopping Servers and Processes

Before you run the Upgrade Assistant to upgrade your 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.

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](#).

 **Note:**

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 System Components**

To stop 11g system components, such as Oracle HTTP Server, use the `opmnctl` script:

 **Note:**

If the Oracle HTTP server is shared with other services, then you can choose *not* to stop the Oracle HTTP server.

- (UNIX) `OHS_INSTANCE_HOME/bin/opmnctl stopall`
- (Windows) `OHS_INSTANCE_HOME\bin\opmnctl stopall`

You can stop system components in any order.

**Step 2: 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')
```

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

### Step 3: 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 following 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 a 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')
```

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

### Step 4: Stop Node Manager

To stop Node Manager, run the following command:

```
kill $(ps -ef | grep nodemanager | | awk '{print $2}')
```

## 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.

- [Identifying Existing Schemas Available for Upgrade](#)  
This optional task enables you to review the list of available schemas before you begin the upgrade by querying the schema version registry. The registry contains schema information such as version number, component name and ID, date of creation and modification, and custom prefix.



- [Starting the Upgrade Assistant](#)  
Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12c (12.2.1.3.0). Oracle recommends that you run the Upgrade Assistant as a non-SYSDBA user, completing the upgrade for one domain at a time.
- [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 task enables you to review the list of available schemas before you begin the upgrade by querying the schema version registry. The registry contains schema information such as version number, component name and ID, date of creation and modification, and custom prefix.

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 ORDER BY MRC_NAME, COMP_ID;
```

2. Examine the report that is generated.

If an upgrade is not needed for a schema, the `schema_version_registry` table retains the schema at its pre-upgrade version.

3. Note the schema prefix name that was used for your existing schemas. You will use the same prefix when you create new 12c schemas.

 **Notes:**

- If you used an OID-based policy store in 11g, 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 12c (12.2.1.3.0). Do not attempt to upgrade a domain that includes components that are not yet available for upgrade to 12c (12.2.1.3.0).

## Starting the Upgrade Assistant

Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12c (12.2.1.3.0). Oracle recommends that you run the Upgrade Assistant as a non-SYSDBA user, completing the upgrade for one domain at a time.

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 ensure that UTF-8 is used by the JVM, use the JVM option -  
`Dfile.encoding=UTF-8`.

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`

 **Note:**

In the above command, `ORACLE_HOME` refers to the 12c (12.2.1.3.0) Oracle Home.

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

# Upgrading Oracle Identity Manager Schemas Using the Upgrade Assistant

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

## Note:

- If the pre-upgrade environment has Audit schema (IAU), you must first upgrade Audit schema only, using the **Individually Selected Schema** option on the Selected Schemas screen, and selecting **Oracle Audit Services** schema. Ensure that you select the appropriate IAU schema from the list of available IAU schemas. The upgrade assistant will not detect the corresponding IAU schema from the provided domain directory automatically. Hence, you must select it manually. Once the IAU schema is upgraded, run the Upgrade Assistant again to upgrade the remaining schemas using the **All Schema Used by a domain** option on the Selected Schemas screen.
- If there is no Audit schema (IAU) in your pre-upgrade environment, use the **All Schema Used by a Domain** option on the Selected Schemas screen and proceed.
- To check whether the pre-upgrade environment has the IAU schema, run the following SQL command using the user with sysdba privileges:

```
select username from dba_users where username like '%IAU%';
```

This command lists the IAU schemas available in your configured database.

## Note:

For SSL enabled setup, it is mandatory to run the Repository Creation Utility (RCU) to upgrade the existing schemas. For more information, see [Creating the Required 12c Schemas Using RCU \(Optional\)](#). For non-SSL enabled setup, running RCU to upgrade schemas is optional.

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 Selected Schemas 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 12c (12.2.1.3.0) components. Do not upgrade schemas that are currently being used to support components that are not included in Oracle Fusion Middleware 12c (12.2.1.3.0).

- **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 your OIM database has only the SSL port open, 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.

Click **Next**.

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.

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 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.

 **Note:**

The component ID or schema name is changed for UCSUMS schema as of release 12.1.2, which means the Upgrade Assistant does not automatically recognize the possible schemas and display them in a drop-down list. You must manually enter the name in a text field. The name can be either *prefix\_ORASDPM* or *prefix\_UMS*, depending on the starting point for the upgrade.

**11g to 12c Upgrades Only:** The UCSUMS schema is not auto-populated. Enter *prefix\_ORASDPM* as the user. The upgrade environment uses *\_ORASDPM* as the schema name, whereas in the 12c environment it is referred to as *\_UMS*.

6. 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.

7. 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.

8. 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**.

9. If the upgrade is successful: On the Upgrade Success screen, 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:

```
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 ORDER BY MRC_NAME, COMP_ID ;
```

In the query result:

- 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 12.2.1.3.0.

 **Note:**

However, that not all schema versions will be updated. Following are the schemas that do not require an upgrade to this release and will retain their pre-upgrade version number:

```
BIPLATFORM
IAU
IAU_APPEND
IAU_VIEWER
OPSS
UCSUMS
WLS
```

- 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.

## About Reconfiguring the Domain

Run the Reconfiguration Wizard to reconfigure your domain component configurations to 12c (12.2.1.3.0).

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:

- 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.

- If the installation that you're upgrading does not use Oracle Access Manager (OAM), then you must edit two files to prevent the Reconfiguration Wizard from attempting to update the nonexistent OAM Infrastructure schema, which causes the upgrade to fail.

Comment out the lines in your `$DOMAIN_HOME/init-info/domain-info.xml` that are similar to this example:

Where, `DOMAIN_HOME` is the Administrator server domain home.

```
<!--extention-template-ref name="Oracle Identity Navigator"
  version="11.1.1.3.0"
  location="/u01/app/oracle/product/fmw/iam111130/common/
templates/applications/oracle.oinav_11.1.1.3.0_template.jar"
symbol="" /-->
<!--install-comp-ref name="oracle.idm.oinav"
version="11.1.1.3.0"

symbol="oracle.idm.oinav_11.1.1.3.0_iam111130_ORACLE_HOME"
  product_home="/u01/app/oracle/product/fmw/iam111130"/-->
```

and similarly comment out the lines in `$DOMAIN_NAME/config/config.xml` that are similar to this example:

```
<!--app-deployment>
  <name>oinav#11.1.1.3.0</name>
  <target>AdminServer</target>
  <module-type>ear</module-type>

  <source-path>/u01/app/oracle/product/fmw/iam111130/oinav/
modules/oinav.ear_11.1.1.3.0/oinav.ear</source-path>
  <deployment-order>500</deployment-order>
  <security-dd-model>DDOnly</security-dd-model>
  <staging-mode>nostage</staging-mode>
</app-deployment-->
```

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.

Follow these instructions to reconfigure the existing domain using the Reconfiguration Wizard. See *Reconfiguring WebLogic Domains in Upgrading Oracle WebLogic Server*.

- [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.

To create a backup of the Administration server domain directory:

1. Copy the source domain to a separate location to preserve the contents.  
(Windows) `copy /Oracle/Middleware/user_projects/domains to /Oracle/Middleware/user_projects/domains_backup.`  
(UNIX) `cp -rf mydomain mydomain_backup`
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](#).

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 environment variable `WLS_ALTERNATIVE_TYPES_DIR` using 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.

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:

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

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

So, first, change the cache directory by setting the environment variable `CONFIG_JVM_ARGS`.

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

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.

 **Note:**

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 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 12c (12.2.1.3.0) is 1.8.0\_131 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 11g datasource, the reconfiguration will preserve the existing values. For new datasources where the schema was created for 12c 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 11g, and your database has `_OPSS` or `_IAU` 11g database schemas, you must manually enter database connection details for those schemas. These schemas were not required in 11g 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.

6. 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**.
7. 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.
8. 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**.

9. 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.

10. 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**.

11. 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

After reconfiguring the domain, 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 12c (12.2.1.3.0). Oracle recommends that you run the Upgrade Assistant as a non-SYSDBA user, completing the upgrade for one domain at a time.
- [Upgrading Oracle Identity Manager Domain Component Configurations](#)  
Navigate through the screens in the Upgrade Assistant to upgrade component configurations in the WebLogic domain.
- [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 12.2.1.3.0.

## Starting the Upgrade Assistant

Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12c (12.2.1.3.0). Oracle recommends that you run the Upgrade Assistant as a non-SYSDBA user, completing the upgrade for one domain at a time.

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 ensure that UTF-8 is used by the JVM, use the JVM option -Dfile.encoding=UTF-8.

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`



### Note:

In the above command, `ORACLE_HOME` refers to the 12c (12.2.1.3.0) Oracle Home.

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 3-5 Upgrade Assistant Command-Line Parameters**

Parameter	Required or Optional	Description
-readiness	Required for readiness checks <b>Note:</b> 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 <code>-examine</code> 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 <i>silent mode</i> (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 <code>-readiness</code> parameter.



**Table 3-5 (Cont.) Upgrade Assistant Command-Line Parameters**

Parameter	Required or Optional	Description
<code>-logLevel attribute</code>	Optional	<p>Sets the logging level, specifying one of the following attributes:</p> <ul style="list-style-type: none"> <li>• TRACE</li> <li>• NOTIFICATION</li> <li>• WARNING</li> <li>• ERROR</li> <li>• INCIDENT_ERROR</li> </ul> <p>The default logging level is NOTIFICATION.</p> <p>Consider setting the <code>-logLevel TRACE</code> 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 <code>-logLevel TRACE</code> is used.</p>
<code>-logDir location</code>	Optional	<p>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.</p> <p>The default locations are:</p> <p>(UNIX)</p> <pre>ORACLE_HOME/  oracle_common/upgrade/  logs  ORACLE_HOME/  oracle_common/upgrade/  temp</pre> <p>(Windows)</p> <pre>ORACLE_HOME\oracle_commo  n\upgrade\logs  ORACLE_HOME\oracle_commo  n\upgrade\temp</pre>
<code>-help</code>	Optional	Displays all of the command-line options.

## Upgrading Oracle Identity Manager Domain Component Configurations

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

After running the Reconfiguration Wizard to reconfigure the WebLogic domain to 12c (12.2.1.3.0), you must 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, enter the WebLogic domain directory path.  
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. If there are remote managed servers hosting User Messaging Services (UMS) configuration files: On the UMS Configuration screen, provide the credentials to these servers so that the Upgrade Assistant can access the configuration files.

 **Note:**

You may need to manually copy the UMS configuration files if the Upgrade Assistant is unable to locate them. See Error while Copying User Messaging Service (UMS) Configuration Files.

6. On the Old (that is, 11g) OIM Home Location screen, select **11g Source**, and specify the absolute path to the 11.1.2.3.0 OIM Oracle Home, which is `ORACLE_HOME/Oracle_IDM`.

Click **Next**.

7. 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.

8. 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.

9. 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**.

10. 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 `NEW_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 12.2.1.3.0.

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

To sign in to the Administration Console in an EDG deployment, see [Validating the Virtual Server Configuration and Access to the Consoles](#).

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

 **Note:**

- After upgrade, ensure you run the administration tools from the new 12c 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.
- In the site-specific configuration, the WebLogic and EM consoles must be accessible with the URLs either directly or through proxy URLs.

## Post-Upgrade Task

After upgrading from 11g to 12c, you need to copy any custom configuration present in your 11g Middleware home to the 12c Oracle home.

- If you have scheduled jobs with parameters referring to the 11g Middleware home, then you need to update them to the corresponding 12c Oracle home.
- To preserve customized configuration data (if present), copy the contents from standard directories such as `XLIntegrations` and `connectorResources` under the 11g Middleware home to the corresponding directories under the 12c Oracle home.
- [Upgrading SOA Composites](#)  
If your starting point is Oracle Identity Manager 11.1.1.x.x, you must manually upgrade custom composites that you have built.

## Upgrading SOA Composites

If your starting point is Oracle Identity Manager 11.1.1.x.x, you must manually upgrade custom composites that you have built.

Complete the following steps to upgrade SOA composites:

1. Open the SOA composite project in JDeveloper (Use Jdeveloper 11.1.1.9.0).
2. Open `ApprovalTask.task` file in designer mode.
3. Select **General**.
4. Change **Owner** to **Group, SYSTEM ADMINISTRATORS, STATIC**.
5. Select **Outcomes lookup**.  
An **Outcomes Dialog** opens.
6. Select **Outcomes Requiring Comment**.
7. Select **Reject** and click **OK**.
8. Click **OK** again.
9. Select **Notification**.
10. Click on the update icon under **Notification**.  
Update any old URLs in notification with the corresponding new URL in 11.1.2.3.0.

Following is an example notification content:

```
A <%/task:task/task:payload/task:RequestModel%> request has been
assigned to you for approval. <BR><BR>
Request ID: <%/task:task/task:payload/task:RequestID%> <BR>
Request type: <%/task:task/task:payload/task:RequestModel%> <BR>
<BR>
Access this task in the
<A
style="text-decoration: none;" href=<%substring-before(/
task:task/task:payload/task:url, "/workflowservice/CallbackService" )
%>/identity/faces/home?tf=approval_details
>
Identity Self Service
</A>
application or take direct action using the links below. Approvers
are required to provide a justification when rejecting the request
```

11. Click **Advanced**.
12. Deselect **Show worklist/workspace URL in notifications**.  
Provide the URL to Pending Approvals in identity application as listed in the example in [step 10](#).
13. Repeat [step 1](#) to [step 12](#) for other human tasks, if any, in the composite, and then save your work.
14. Right click **Project** and select **Deploy > Deploy to Application Server**.
15. Provide revision ID.  
Select **Mark revision as default** and **Overwrite any existing composite with same revision ID**.

 **Note:**

You can also deploy the composites with different revision ID. In that case, you have to modify all approval policies using this composite.

16. Select your application server connection, if it already exists, and click **Next**.  
Create an application server connection if it does not exist.
17. Click **Next**.
18. Click **Finish**.
19. Repeat the procedure for the remaining custom composites.

## Copying folders to the 12c Oracle Home

When upgrading to 12c, you must manually copy some folders to the 12c Oracle Home, if those folders are having file system dependent data.

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

Run the following command:

```
cp -r 11g_MW_HOME/<product_idm>/server/plugins/* ORACLE_HOME/  
<product_idm>/server/plugins/
```

Where, *ORACLE\_HOME* is the 12c Oracle Home.

## Starting the Servers

After you upgrade Oracle Identity Manager, start the servers.

You must start the servers in the following order:

1. Start the Administration Server.

If Node manager is configured, do not start the Node Manager.

2. From the terminal, start the Oracle SOA Suite managed server with the Administration Server URL, and the BPM property set to `TRUE`.

For example:

```
./startManagedWebLogic.sh <SOA_Managed_server> t3://  
weblogic_admin_host:weblogic_admin_port -Dbpm.enabled=true
```

### Note:

- For first boot, Oracle SOA Suite managed server must be started manually by using the command in the above example.
- In SSL environment, when starting managed servers for the first time for bootstrap, provide the non-SSL port number of the Administration Server.

3. Once the SOA server is in running state, from the terminal, start the Oracle Identity Manager Managed Server with the Administration Server URL.

This time, OIM bootstrap process will be executed, and after successful bootstrap, OIM Managed Server will be shut down automatically.

For example:

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

### Note:

As in the previous step, provide the non-SSL port number of the Administration Server.

4. Shut down the SOA Managed Server and the Administration Server. For information about stopping the servers and processes, see [Stopping Servers and Processes](#).

- [Starting Servers and Processes](#)

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

## 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 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

Starting the Node Manager in the Administration Server domain home:

- (UNIX) `nohup ./startNodeManager.sh > DOMAIN_HOME/nodemanager/nodemanager.out 2>&1 &`
- (Windows) `nohup .\startNodeManager.sh > DOMAIN_HOME\nodemanager\nodemanager.out 2>&1 &`

### 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

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:**

- 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 11g 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.

## 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, replace the 12c (12.2.1.3.0) `ORACLE_HOME/idm/designconsole/config/xlconfig.xml` file with the 11.1.2.3.0 `ORACLE_HOME/Oracle_IDM1/designconsole/config/xlconfig.xml` file.

## Completing the Post-Upgrade Tasks for SSL Enabled Setup

If you are upgrading an Oracle Identity Manager SSL enabled setup, you must perform the required post-upgrade tasks to complete the upgrade process.

Complete the following tasks if you have upgraded an SSL enabled setup:

1. Changes done for SSL settings in `setDomainEnv.sh`, `startWeblogic.sh`, `startManagedWeblogic.sh`, and `datasources` are lost after upgrade. Re-do all of the changes.
2. Start the WebLogic Administration Server. To start the Administration Server, use the `startWebLogic` script:
  - (UNIX) `DOMAIN_HOME/bin/startWebLogic.sh`
  - (Windows) `DOMAIN_HOME\bin\startWebLogic.cmd`

Where, `DOMAIN_HOME` is the Administration domain.

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

3. Make necessary changes to the following newly created `datasources`, for SSL settings:

- LocalSvcTblDataSource
- opss-audit-DBDS
- opss-audit-viewDS
- opss-data-source
- WLSSchemaDataSource

For information about updating the newly created datasources, see [Updating Datasource oimOperationsDB Configuration in \*Administering Oracle Identity Governance\*](#)

4. In case of Customer Identity and Java Standard Trust, import your identity trust certificate to the new JDK home. The 12c (12.2.1.3.0) uses `jdk1.8.0_131`. To import the identity trust certificate to the new JDK home, use the following command:

```
./keytool -importcert -alias startssl -keystore JAVA_HOME/jre/lib/  
security/cacerts -storepass changeit -file supportcert.pem
```

5. Verify that all of the SSL configuration changes including the SSL port related changes done in 11g (pre upgrade), are present post upgrade. If the changes are lost, you must redo them post upgrade. Some of the SSL configuration changes include:

- OimFrontEndURL
- backOfficeURL
- SOA Server URL
- ForeignJNDIProvider-SOA

For more information about configuring SSL for Oracle Identity Governance, see [Updating Oracle Identity Governance in \*Administering Oracle Identity Governance\*](#).

## Installing Standalone Oracle BI Publisher

When you upgrade Oracle Identity Manager 11.1.2.3.0 to Oracle Identity Governance 12c (12.2.1.3.0), the embedded Oracle BI Publisher present in the 11.1.2.3.0 deployment, is removed. Therefore, you must install a new standalone Oracle BI Publisher 12c (12.2.1.3.0) post upgrade, for configuring the Oracle Identity Governance reports.

For information about installing and configuring Oracle BI Publisher 12c (12.2.1.3.0), see [Installing and Configuring Oracle BI Publisher in \*Developing and Customizing Applications for Oracle Identity Governance\*](#).

For information about integrating standalone Oracle BI Publisher with Oracle Identity Governance 12c (12.2.1.3.0), see [Integrating Standalone BI Publisher with Oracle Identity Governance in \*Developing and Customizing Applications for Oracle Identity Governance\*](#).

## Tuning Application Module for User Interface

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

See Tuning Application Module (AM) for User Interface in *Oracle Fusion Middleware Tuning Performance*.

# 4

## Upgrading Oracle Identity Manager Highly Available Environments

Describes the process of upgrading an Oracle Identity Manager highly available environment from 11g Release 2 (11.1.2.3.0) to Oracle Identity Governance 12c (12.2.1.3.0).

### Note:

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

### Topics

- [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.
- [Completing the Pre-Upgrade Tasks for Oracle Identity Manager](#)  
Complete the pre-upgrade tasks described in this section before you upgrade Oracle Identity Manager.
- [Creating 12c Oracle Home Folder on OIMHOST1 and OIMHOST2](#)  
Create a folder for 12c Oracle Home on both OIMHOST1 and OIMHOST2.
- [Installing Product Distributions on OIMHOST1 and OIMHOST2](#)  
Install the 12c binaries onto OIMHOST1 and OIMHOST2 or onto shared storage accessible by both. If you are using redundant binaries ensure you install into each of the redundant locations
- [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.
- [Creating the Required 12c Schemas Using RCU \(Optional\)](#)  
When upgrading from 11g, you must create the required 12c schemas. If your setup has non-SSL ports open, you can use the Upgrade Assistant to create schemas by using the default schema settings. In case of SSL enabled setup, you can use the Repository Creation Utility (RCU) to create customized schemas. This procedure describes how to create schemas using the RCU. Information about using the Upgrade Assistant to create schemas is covered in the upgrade procedures.

- [Stopping Servers and Processes](#)  
Before you run the Upgrade Assistant to upgrade your 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.
- [Upgrading Schemas on OIMHOST1](#)  
Upgrade all of the necessary schemas for Oracle Identity Manager, on OIMHOST1 by using the Upgrade Assistant.
- [Reconfiguring the Domain on OIMHOST1](#)  
Run the Reconfiguration Wizard on OIMHOST1 to reconfigure your domain component configurations to 12c (12.2.1.3.0).
- [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.
- [Starting the Servers](#)  
After you upgrade Oracle Identity Manager, start the servers.
- [Replicating the Domain Configurations on each OIMHOST](#)  
Replicate the domain configurations on OIMHOST2. This involves packing the upgraded domain on OIMHOST1 and unpacking it on OIMHOST2.
- [Copying oracle.iam.ui.custom-dev-starter-pack.war to the 12c \(12.2.1.3.0\) Oracle Home](#)  
After you run the Upgrade Assistant for config upgrade, manually copy the `oracle.iam.ui.custom-dev-starter-pack.war` file from backup of 11g Oracle home to 12c (12.2.1.3.0) Oracle home.
- [Starting the Servers on OIMHOST1 and OIMHOST2](#)  
After you upgrade Oracle Identity Manager on both OIMHOST1 and OIMHOST2, start the servers.
- [Upgrading Oracle Identity Manager Design Console](#)  
Upgrade the Oracle Identity Manager Design Console after you upgrade the Oracle Identity Manager (OIM) domain component configurations.
- [Completing the Post-Upgrade Tasks for SSL Enabled Setup](#)  
If you are upgrading an Oracle Identity Manager SSL enabled setup, you must perform the required post-upgrade tasks to complete the upgrade process.
- [Installing Standalone Oracle BI Publisher](#)  
When you upgrade Oracle Identity Manager 11.1.2.3.0 to Oracle Identity Governance 12c (12.2.1.3.0), the embedded Oracle BI Publisher present in the 11.1.2.3.0 deployment, is removed. Therefore, you must install a new standalone Oracle BI Publisher 12c (12.2.1.3.0) post upgrade, for configuring the Oracle Identity Governance reports.

## 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.

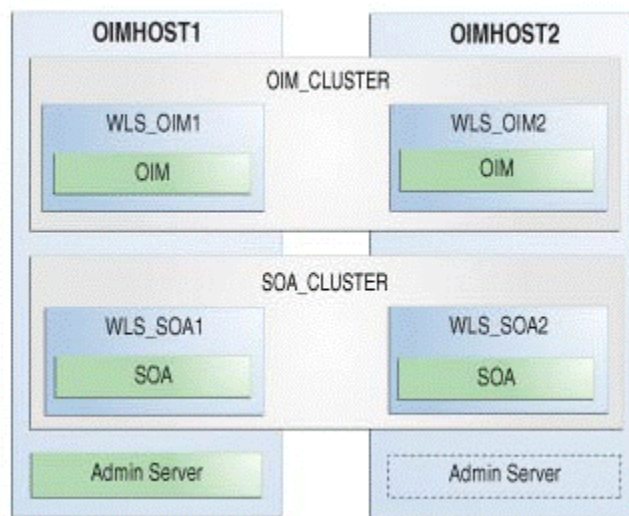
## Upgrade Topology

The following topology shows the Oracle Identity Manager cluster set up that can be upgraded to 12c (12.2.1.3.0) by following the procedure described in this chapter.

 **Note:**

As required, you can upgrade OHS independently of this OIM upgrade process.

**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.

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

Task	Description
<p><b>Required</b> If you have not done so already, review the introductory topics in this guide and complete the required pre-upgrade tasks.</p>	<p>See:</p> <ul style="list-style-type: none"> <li>• <a href="#">Introduction to Upgrading Oracle Identity Manager to 12c (12.2.1.3.0)</a></li> <li>• <a href="#">Pre-Upgrade Requirements</a></li> </ul>
<p><b>Required</b> Complete the necessary pre-upgrade tasks specific to Oracle Identity Manager.</p>	<p>See <a href="#">Completing the Pre-Upgrade Tasks for Oracle Identity Manager</a>.</p>
<p><b>Required</b> Create the 12c Middleware Home Folder on both OIMHOST1 and OIMHOST2, so that you can use the location for installing the product distributions.</p>	<p>See <a href="#">Creating 12c Middleware Home Folder on OIMHOST1 and OIMHOST2</a>.</p>
<p><b>Required</b> Install Oracle SOA Suite12c (12.2.1.3.0) and Oracle Identity Manager12c (12.2.1.3.0) in the new Oracle home.</p>	<p>See <a href="#">Installing Product Distributions on OIMHOST1 and OIMHOST2</a>.</p>
<p><b>Required</b> Run a pre-upgrade readiness check.</p>	<p>See <a href="#">Running a Pre-Upgrade Readiness Check</a>.</p>
<p><b>Required</b> Start the Repository Creation Utility (RCU) to create the required 12c database schemas.</p> <p><b>Note:</b> This step is not required for non-SSL setup, as the Upgrade Assistant creates the necessary 12c schemas during the upgrade process.</p> <p>For SSL enabled setup, you must run the RCU to create the necessary 12c schemas.</p>	<p>The schemas you create will vary depending on your existing schema configuration.</p> <p>See <a href="#">Creating the Required 12c Schemas with the RCU</a>.</p>
<p><b>Required</b> Shut down the 11g servers. This includes the Administration Server, Managed Servers, Node Manager, and system components like Oracle HTTP Server.</p> <p>Ensure that the Database is up during the upgrade.</p>	<p>See <a href="#">Stopping Servers and Processes</a>.</p>
<p><b>Required</b> Upgrade the necessary schemas on OIMHOST1.</p>	<p>See <a href="#">Upgrading Schemas on OIMHOST1</a>.</p>
<p><b>Required</b> Reconfigure the Oracle Identity Manager domain on OIMHOST1.</p>	<p>See <a href="#">Reconfiguring the Domain on OIMHOST1</a>.</p>
<p><b>Required</b> Upgrade the Oracle Identity Manager configurations on both OIMHOST1, using the Upgrade Assistant.</p>	<p>The Upgrade Assistant is used to update the reconfigured domain's component configurations.</p> <p>See <a href="#">Upgrading Domain Component Configurations on OIMHOST1 and OIMHOST2</a>.</p>
<p><b>Required</b> Replicate the domain configurations on OIMHOST2.</p>	<p>This includes packing the domain on OIMHOST1 and unpacking it on OIMHOST2.</p> <p>See <a href="#">Replicating the Domain Configurations on OIMHOST2</a>.</p>

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

Task	Description
<b>Required</b> Copy the <code>oracle.iam.ui.custom-dev-starter-pack.war</code> from 11g Middleware Home to 12c Middleware Home.	See <a href="#">Copying oracle.iam.ui.custom-dev-starter-pack.war to the 12c (12.2.1.3.0) Oracle Home</a> .
<b>Required</b> Start the servers on OIMHOST1 and OIMHOST2.	See <a href="#">Starting the Servers</a> .
<b>Required</b> Upgrade the Oracle Identity Manager Design Console to 12c (12.2.1.3.0).	See <a href="#">Upgrading Oracle Identity Manager Design Console</a> .
<b>Required</b> Perform the post-upgrade tasks for SSL enabled setup. <b>Note:</b> This step is not required for non-SSL setup.	See <a href="#">Completing the Post-Upgrade Tasks for SSL Enabled Setup</a> .
<b>Optional</b> When you upgrade to Oracle Identity Governance 12c (12.2.1.3.0), the embedded Oracle BI Publisher present in the 11.1.2.3.0 deployment is removed. Therefore, you must install a new standalone Oracle BI Publisher 12c (12.2.1.3.0) on OIMHOST1 and OIMHOST2, post upgrade. After you install, integrate it with Oracle Identity Governance 12c (12.2.1.3.0) to configure the Oracle Identity Governance reports.	See <a href="#">Installing Standalone Oracle BI Publisher</a> .
<b>Required</b> Connector server upgrade	After upgrading to 12c (12.2.1.3.0), see <a href="#">Oracle Identity Manager Connectors Certification Matrix</a> to check if any connectors have to be updated.  For more information, see <a href="#">Oracle Identity Governance 12c Connectors</a> .

## Completing the Pre-Upgrade Tasks for Oracle Identity Manager

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

- Updating Server Wallets to Remove MD5 Algorithm**  
 If the existing keystore has a certificate which is invalid with the JDK that is used to install 12c (12.2.1.3.0) binaries, you must generate the keystore and place it in the `DOMAIN_HOME/config/fmwconfig` directory.
- Updating DB Wallets to Remove MD5 Algorithm (For SSL Enabled Setup)**  
 If you have SSL enabled setup, update all of the DB wallets to remove any MD5 algorithms, as 12c (12.2.1.3.0) uses JDK 8 which does not support MD5 algorithm.
- 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 Servers and Database before you proceed with the Oracle Identity Manager upgrade.

## Updating Server Wallets to Remove MD5 Algorithm

If the existing keystore has a certificate which is invalid with the JDK that is used to install 12c (12.2.1.3.0) binaries, you must generate the keystore and place it in the `DOMAIN_HOME/config/fmwconfig` directory.

If the default keystore has MD5 algorithm, then the upgrade readiness check and the examine phase of OIM configuration upgrade will fail.

To verify the validity of the certificate, do the following:

1. Check for the `jdk.certpath.disabledAlgorithms` property in the `JAVA_HOME/jre/lib/security/java.security` file.

For example:

```
jdk.certpath.disabledAlgorithms=MD2, MD5, RSA keySize < 1024
```

2. Check for the certificate algorithm in the existing keystore by doing the following:
  - a. For default keystore, `DOMAIN_HOME/config/fmwconfig/default-keystore.jks`, run the following command from the `JAVA_HOME/jre/bin` directory:

```
./keytool -list -v -keystore DOMAIN_HOME/config/fmwconfig/default-keystore.jks
```

If you are using the custom keystores, that is, `DOMAIN_HOME/config/fmwconfig/name_of_custom_store`, run the following command from the `JAVA_HOME/jre/bin` directory:

```
./keytool -list -v -keystore DOMAIN_HOME/config/fmwconfig/custom_keystore.jks
```

This command displays the keystore data. Enter the keystore password when prompted.

- b. Check for the `Signature algorithm name` field value in the output of the above command. If the value of `Signature algorithm name` field and the `jdk.certpath.disabledAlgorithms` property has MD5 algorithm, then the given keystore will not be valid after upgrade.

If the keystore is not valid after upgrade, the following error is seen in the server logs while executing the request use cases after upgrade, and none of the request use cases will be successful:

```
Caused by: java.security.cert.CertPathValidatorException:
Algorithm
constraints check failed: MD5withRSA
```

3. Generate a certificate signing request from the temporary keystore and load it into `<cert_req>.csr` with the alias `xell` using the `keytool` command:

```
keytool -certreq -alias xell -file /tmp/<cert_req>.csr -
keypass keypassword -keystore /temporary location/default-keystore.jks -
storepass storepassword -storetype jks
```

 **Note:**

The following commands can be used to generate both default keystore and custom keystores.

- `./keytool -genkeypair -keystore new_keystore_location/new_keystore_name.jks -keyalg supported_algorithm_name -sigalg SHA256withRSA -validity validity_period -keypass key_password -storepass keystore_password -alias xell -dname vaild_name -keysize key_size`

For example:

```
./keytool -genkeypair -keystore /scratch/default-keystore.jks
-keyalg RSA -sigalg SHA256withRSA -validity 3600 -
keypass password -storepass password -alias xell -dname
"CN=oimhost,OU=Identity,O=ABC,C=XX" -keysize 2048
```

- `./keytool -exportcert -keystore new_keystore_location/new_keystore_name.jks -rfc -file new_keystore_location/certificate_name.cer`

For example:

```
./keytool -exportcert -keystore /scratch/default-keystore.jks -v
-alias xell -storepass password -rfc -file /scratch/cert.cer
```

- `./keytool -importcert -keystore new_keystore_location/new_keystore_name.jks -alias xeltrusted -file new_keystore_location/certificate_name.cer -storepass keystore_password`

Click **Yes** to confirm the action.

For example:

```
./keytool -importcert -keystore /scratch/default-keystore.jks -
alias xeltrusted -file /scratch/cert.cer -storepass password
```

4. Import the newly generated keystore into the existing keystore `DOMAIN_HOME/config/fmwconfig/default-keystore.jks` by running the following command:

```
./keytool -importkeystore -srckeystore new_keystore_location/new_keystore_name.jks -destkeystore DOMAIN_HOME/config/fmwconfig/
```

```
default-keystore.jks -srcstorepass source_keystore_password -
deststorepass destination_keystore_password -noprompt
```

For example:

```
./keytool -importkeystore -srckeystore /scratch/default-
keystore.jks -destkeystore domain_home/config/fmwconfig/default-
keystore.jks -srcstorepass password -deststorepass password -
noprompt
```

5. Log in to Enterprise Manager console and update the `xell` named CSF key under `oim` map, with the password value which is used above to generate the new key in keystore. In the above example, the password used is `password`.
6. Move the `<export file>.cert` and the `<cert_req>.csr` to the `DOMAIN_HOME/config/fmwconfig/` location.

```
cp /tm/xell.csr DOMAIN_HOME/config/fmwconfig/
cp /tmp/xlserver.cert DOMAIN_HOME/config/fmwconfig/
```

#### Note:

For more information about using the `keytool` command, see [keytool](#) in the *Java Platform, Standard Edition Tools Reference*.

#### Note:

The procedure described in this section for regenerating the `default-keystore.jks` or custom keystore includes self-signed certificates. If CA signed certificate is required, follow the standard process for the same, that is, generate the CSR and import the signed certificates in the keystore.

During bootstrap process in OIM, the `default-keystore.jks` keystore will be configured in Keystore Service (KSS) out-of-the-box. In case of custom keystore, upload the given custom keystore to KSS after completing the upgrade. After you upload the given custom keystore to KSS, restart the servers.

For more information about the Keystore Service commands, see *OPSS Keystore Service Commands in WLST Command Reference for Infrastructure Security*.

## Updating DB Wallets to Remove MD5 Algorithm (For SSL Enabled Setup)

If you have SSL enabled setup, update all of the DB wallets to remove any MD5 algorithms, as 12c (12.2.1.3.0) uses JDK 8 which does not support MD5 algorithm.

### Note:

All these steps in this procedure must be performed on the Database server. That is, on the server where OIM database is installed.

To update the DB wallet, do the following:

1. Create an Oracle Wallet with default trusted certificate using the following command:

```
./orapki wallet create -wallet <trust_wallet_name> -pwd password
```

For example:

```
./orapki wallet create -wallet trust_wallet.p12 -pwd password
```

2. Add a self-signed certificate in the wallet with the distinguished name (DN) as CN=root\_test,C=US using the following command:

```
./orapki wallet add -wallet trust_wallet_name -dn 'dn_name' -keysize 2048 -sign_alg sha256 -self_signed -validity 3650 -pwd password_of_wallet
```

For example:

```
./orapki wallet add -wallet trust_wallet.p12 -dn 'CN=root_test,C=US' -keysize 2048 -sign_alg sha256 -self_signed -validity 3650 -pwd password
```

3. Export the self-signed trust certificate from the Oracle wallet to use it to sign other certificates, using the following command:

```
./orapki wallet export -wallet trust_wallet_name -dn 'dn_name' -cert trust_cert_file_name -pwd password_of_wallet
```

For example:

```
./orapki wallet export -wallet trust_wallet.p12 -dn 'CN=root_test,C=US' -cert wallet_trusted.cert -pwd password
```

4. You already have an Oracle Wallet with User Certificate identified. The user wallet is, *DB\_HOME/bin/user\_wallet.p12*. The DN of this user certificate is CN=Customer,OU=Customer,O=Customer,L=City,ST=NY,C=US. Remove the existing user certificate from this wallet using the following command:

Where, *DB\_HOME* is the server where OIM database is installed.

```
./orapki wallet remove -wallet user_wallet_name -pwd password_of_existing_wallet -dn 'DN_name' -user_cert
```

For example:

```
./orapki wallet remove -wallet user_wallet.p12 -pwd password -dn 'CN=Customer,OU=Customer,O=Customer,L=City,ST=NY,C=US' -user_cert
```

5. You already have an Oracle Wallet with Requested Certificate identified. The user wallet is, `DB_HOME/bin/user_wallet.p12`. The DN of this requested certificate is `CN=Customer,OU=Customer,O=Customer,L=City,ST=NY,C=US`. Remove the existing requested certificate from this wallet using the following command:

```
./orapki wallet remove -wallet user_wallet_name -dn 'DN_name' -
cert_req -pwd password_of_existing_wallet
```

For example:

```
./orapki wallet remove -wallet user_wallet.p12 -dn
'CN=Customer,OU=Customer,O=Customer,L=City,ST=NY,C=US' -cert_req -pwd
password
```

6. You already have an Oracle Wallet with Trust Certificate identified. The user wallet is, `DB_HOME/bin/user_wallet.p12`. The DN of this trust certificate is `CN=root_test,C=US`. Remove the existing trust certificate from this wallet using the following command:

```
./orapki wallet remove -wallet user_wallet_name -pwd password-of-
existing_wallet -dn 'DN_name' -trusted_cert
```

For example:

```
./orapki wallet remove {wallet user_wallet.p12 -pwd password -dn '
CN=root_test,C=US' -trusted_cert
```

7. Add a user certificate in the existing user wallet with a distinguished name as `CN=Customer,OU=Customer,O=Customer,L=City,ST=NY,C=US` using the following command:

```
./orapki wallet add -wallet user_wallet_name -dn 'dn_name' -keysize
2048 -sign_alg sha256 -pwd password_of_existing_wallet
```

For example:

```
./orapki wallet add -wallet user_wallet.p12 -dn
'CN=Customer,OU=Customer,O=Customer,L=City,ST=NY,C=US' -keysize 2048 -
sign_alg sha256 -pwd password
```

8. Export the user certificate request to a file using the following command:

```
./orapki wallet export -wallet user_wallet_name -dn 'dn_name' -request
CSR_file_name -pwd password_of_existing_wallet
```

For example:

```
./orapki wallet export -wallet user_wallet.p12 -dn
'CN=Customer,OU=Customer,O=Customer,L=City,ST=NY,C=US' -request
server_creq.csr -pwd password
```

9. Sign the user certificate request using the trusted wallet that was created above, using the following command:

```
./orapki cert create -wallet trusted_wallet_name-request
CSR_file_name -cert user_cert_file_name sign_alg sha256 -pwd
password_of_exiting_user_wallet
```

For example:

```
./orapki cert create -wallet trust_wallet.p12 -request server_creq.csr
-cert wallet_user.cert -sign_alg sha256 - validity 3650 -pwd
password
```

10. Add the trusted certificate `wallet_trusted.cert` that you created using the above procedure to the wallet, by running the following command:

```
./orapki wallet add -wallet user_wallet_name -trusted_cert -cert
trust_cert_file_name -pwd password_of_exiting_user_wallet
```

For example:

```
./orapki wallet add -wallet user_wallet.p12 -trusted_cert -cert
wallet_trusted.cert -pwd password
```

11. Add the signed user certificate to the Oracle wallet using the following command:

```
./orapki wallet add -wallet user_wallet -user_cert -cert
user_cert_file_name -pwd password_of_exiting_user_wallet

./orapki wallet add -wallet user_wallet.p12 -user_cert -cert
wallet_user.cert -pwd password
```

12. Remove the DB trusted certificate from server keystore. In case of demo identity and demo trust, remove from default-keystore.jks, and in case of custom identity and custom trust, remove it from the custom trust keystore, using the following command:

```
./keytool -delete -alias alias_of_db_cert -keystore custom_trust_store
-storepass password-of-existing-trust-keystore
```

For example:

```
./keytool -delete -alias dbtrusted -keystore DOMAIN_HOME/config/
fmwconfig/custom_trust_store.jks -storepass password
```

13. Import self signed DB certifiacte in trust wallet using the following command:

```
keytool -import -trustcacerts -alias <alias_of_db_cert> -noprompt
-keystore custom_trust_store -file DB_Trust_cert_file -storepass
password_of_existing_trust_keystore
```

For example:

```
keytool -import -trustcacerts -alias dbtrusted -noprompt -
keystore DOMAIN_HOME/config/fmwconfig/custom_trust_store.jks -file /
DB_HOME/bin/wallet_trusted.cert -storepass password
```

## 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` 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. Run the following command to increase the limit of open files, so that you do not run into memory issues:

```
limit maxproc 16384
```

To ensure that the settings persist after a reboot, add the `limit maxproc 16384` entry to `/etc/security/limits.conf` or `/etc/security/limits.d` file.

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 Servers and Database before you proceed with the Oracle Identity Manager upgrade.

To enable non-SSL ports for servers, complete the following steps:

1. Log in to the WebLogic Server Administration Console.
2. Click **Environment** > **Servers** > and select the required admin server.
3. On the **Settings for Server** page, click the **Configuration** tab, and then click **General**.
4. Click **Lock & Edit**.
5. Select **Listen Port Enabled**. The default port is 14000.
6. Repeat the [step 1](#) through [step 5](#) for each required server in the domain.



### Note:

After you complete the upgrade, you can undo these changes as required.

For database: Ensure that database listener is listening on the same TCP port for database servers that you provided to upgrade assistant as parameters. For more information, see [Enabling SSL for Oracle Identity Governance DB](#).

## Creating 12c Oracle Home Folder on OIMHOST1 and OIMHOST2

Create a folder for 12c Oracle Home on both OIMHOST1 and OIMHOST2.

It is recommended that you have the identical directory structure on OIMHOST1 and OIMHOST2.

For example:

```
/home/Oracle/product/ORACLE_HOME
```

## Installing Product Distributions on OIMHOST1 and OIMHOST2

Install the 12c binaries onto OIMHOST1 and OIMHOST2 or onto shared storage accessible by both. If you are using redundant binaries ensure you install into each of the redundant locations

Install the following products on both OIMHOST1 and OIMHOST2:

- Oracle Fusion Middleware Infrastructure 12c (12.2.1.3.0)
- Oracle SOA Suite 12c (12.2.1.3.0)
- Oracle Identity Manager 12c (12.2.1.3.0)

 **Note:**

If you have redundant *Oracle\_Home* installations, then install the binaries into each of the redundant locations.

- [Installing Product Distributions](#)  
Before beginning your upgrade, download Oracle Fusion Middleware Infrastructure, Oracle SOA Suite, and Oracle Identity Manager 12c (12.2.1.3.0) distributions on the target system and install them using Oracle Universal Installer.

## Installing Product Distributions

Before beginning your upgrade, download Oracle Fusion Middleware Infrastructure, Oracle SOA Suite, and Oracle Identity Manager 12c (12.2.1.3.0) distributions on the target system and install them using Oracle Universal Installer.

 **Note:**

The 12c binaries are installed in a different location from the previous 11g binaries. You can install 12c 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 quick installer (*fmw\_12.2.1.3.0\_idmquickstart\_generic.jar*). The quick installer installs the Infrastructure, Oracle SOA Suite, and Oracle Identity Manager 12c (12.2.1.3.0) 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 Quick 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 12c (12.2.1.3.0) separately. To do this, complete the following steps:

1. Sign in to the target system.
2. Download the following from [Oracle Technical Resources](#) or [Oracle Software Delivery Cloud](#) to your target system:



- Oracle Fusion Middleware Infrastructure  
(fmw\_12.2.1.3.0\_infrastructure\_generic.jar)
  - Oracle SOA Suite (fmw\_12.2.1.3.0\_soa\_generic.jar)
  - Oracle Identity Manager (fmw\_12.2.1.3.0\_idm\_generic.jar)
3. Change to the directory where you downloaded the 12c (12.2.1.3.0) product distribution.
  4. Start the installation program for Oracle Fusion Middleware Infrastructure:
    - (UNIX) `JAVA_HOME/bin/java -jar fmw_12.2.1.3.0_infrastructure_generic.jar`
    - (Windows) `JAVA_HOME\bin\java -jar fmw_12.2.1.3.0_infrastructure_generic.jar`
  5. 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.

6. On the Welcome screen, review the information to make sure that you have met all the prerequisites. Click **Next**.
7. 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**.

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

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

9. On the Installation Type screen, select the following:
  - For Infrastructure, select **Fusion Middleware Infrastructure**
  - For Oracle SOA Suite, select **Oracle SOA Suite**

- For Oracle Identity Manager, select **Oracle Identity and Access Management**

Click **Next**.

10. 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).

11. 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.

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

13. 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.

14. 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 12c (12.2.1.3.0), run the following installer:

- (UNIX) `JAVA_HOME/bin/java -jar fmw_12.2.1.3.0_soa_generic.jar`
- (Windows) `JAVA_HOME\bin\java -jar fmw_12.2.1.3.0_soa_generic.jar`

For installing Oracle Identity Manager 12c (12.2.1.3.0), run the following installer:

- (UNIX) `JAVA_HOME/bin/java -jar fmw_12.2.1.3.0_idm_generic.jar`
- (Windows) `JAVA_HOME\bin\java -jar fmw_12.2.1.3.0_idm_generic.jar`

#### Note:

By using the opatch tool, apply the latest recommended patchsets from Oracle Support. Complete only the binary installation of patchsets and follow any post-patch steps after the upgrade process is complete. This provides the latest known fixes for upgrade process, if any.

For more information about installing Oracle Identity Manager 12c (12.2.1.3.0), 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 12c 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 4-2 Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
-readiness	Required for readiness checks <b>Note:</b> 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 <i>silent mode</i> (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 <i>attribute</i>	Optional	Sets the logging level, specifying one of the following attributes: <ul style="list-style-type: none"> <li>• TRACE</li> <li>• NOTIFICATION</li> <li>• WARNING</li> <li>• ERROR</li> <li>• INCIDENT_ERROR</li> </ul> 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 4-2 (Cont.) Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
<code>-logDir location</code>	Optional	<p>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.</p> <p>The default locations are:</p> <p>(UNIX)</p> <pre>ORACLE_HOME/ oracle_common/upgrade/ logs ORACLE_HOME/ oracle_common/upgrade/ temp</pre> <p>(Windows)</p> <pre>ORACLE_HOME\oracle_commo n\upgrade\logs ORACLE_HOME\oracle_commo n\upgrade\temp</pre>
<code>-help</code>	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.

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 11.1.2.3.0 Administrator server domain directory.
- Specify schema credentials to connect to the selected schema: **Database Type**, **DBA User Name**, and **DBA Password**. 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**.

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**.

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 4-3 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 <i>ORACLE_HOME</i> /oracle_common/ upgrade/logs	The directory location of the generated log file.	No action required.
Readiness report location <i>ORACLE_HOME</i> /oracle_common/ upgrade/logs	The directory location of the generated readiness report.	No action required.



Table 4-3 (Cont.) Readiness Report Elements

Report Information	Description	Required Action
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 Tue May 30 11:15:52 EDT 2016
Log file is located at: NEW_ORACLE_HOME/oracle_common/upgrade/logs/
ua2016-05-30-11-14-06AM.log
Readiness Check Report File: NEW_ORACLE_HOME/oracle_common/upgrade/logs/
readiness2016-05-30-11-15-52AM.txt
```

```
Starting readiness check of components.
```

```
Oracle Metadata Services
```

```
Starting readiness check of Oracle Metadata Services.
```

```
Schema User Name: DEV11_MDS
```

```
Database Type: Oracle Database
```

```
Database Connect String: machinename@yourcompany.com
```

```
VERSION Schema DEV11_MDS is currently at version 12.1.1.1.0.
```

```
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
EXCEPTION Schema is missing a required procedure:
GETREPOSITORYFEATURES
Completed schema test: TEST_REQUIRED_PROCEDURES --> Test that the
schema contains all the required stored procedures +++ FAIL
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
Completed index test for table MDS_ATTRIBUTES: TEST_REQUIRED_INDEXES
--> Test that the table contains all the required indexes +++ PASS
Starting index test for table MDS_COMPONENTS: TEST_REQUIRED_INDEXES
--> Test that the table contains all the required indexes
Completed index test for table MDS_TXN_LOCKS: TEST_REQUIRED_INDEXES
--> Test that the table contains all the required indexes +++ PASS
Starting schema test: TEST_REQUIRED_TRIGGERS Test that the schema
has all the required triggers
Completed schema test: TEST_REQUIRED_TRIGGERS --> Test that the
schema has all the required triggers +++ 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_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_UNEXPECTED_PROCEDURES Test that the
schema does not contain any unexpected stored procedures
Completed schema test: TEST_UNEXPECTED_PROCEDURES --> Test that the
schema does not contain any unexpected stored procedures +++ PASS
Starting schema test: TEST_UNEXPECTED_VIEWS Test that the schema
does not contain any unexpected views
Completed schema test: TEST_UNEXPECTED_VIEWS --> Test that the
schema does not contain any unexpected views +++ PASS
Starting index test for table MDS_ATTRIBUTES:
TEST_UNEXPECTED_INDEXES --> Test that the table does not contain any
unexpected indexes
Completed index test for table MDS_ATTRIBUTES:
TEST_UNEXPECTED_INDEXES --> Test that the table does not contain any
unexpected indexes +++ PASS
Completed index test for table MDS_LABELS: TEST_UNEXPECTED_INDEXES
--> Test that the table does not contain any unexpected indexes +++ PASS
Starting index test for table MDS_LARGE_ATTRIBUTES:
TEST_UNEXPECTED_INDEXES --> Test that the table does not contain any
unexpected indexes
Starting schema test: TEST_UNEXPECTED_TRIGGERS Test that the
schema does not contain any unexpected triggers
Completed schema test: TEST_UNEXPECTED_TRIGGERS --> Test that the
schema does not contain any unexpected triggers +++ 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 MDS_ATTRIBUTES:
TEST_COLUMN_DATATYPES_V2 --> Test that all table columns have the
proper datatypes
Completed datatype test for table MDS_ATTRIBUTES:
TEST_COLUMN_DATATYPES_V2 --> Test that all table columns have the
proper datatypes +++ PASS
Starting datatype test for table MDS_COMPONENTS:
TEST_COLUMN_DATATYPES_V2 --> Test that all table columns have the
proper datatypes
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_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 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 11.2.0.3.0 - 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:
FAILURE.
```

If you are running the 12.1.3.0 version of Oracle Fusion Middleware IAU Schemas, and those schemas were upgraded from 11g (11.1.1.7 and later) or 12c (12.1.2.0), your readiness check may fail with the following error:



**Note:**

This is not applicable for Oracle Identity Manager.

```
Starting index test for table IAU_COMMON: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes
INFO Audit schema index DYN_EVENT_CATEGORY_INDEX in table IAU_COMMON is
```

missing the required columns or index itself is missing. This maybe caused by a known issue, anyway, this missing index will be added in 12.2.2 upgrade.

INFO Audit schema index DYN\_EVENT\_TYPE\_INDEX in table IAU\_COMMON is missing the required columns or index itself is missing. This maybe caused by a known issue, anyway, this missing index will be added in 12.2.2 upgrade.

INFO Audit schema index DYN\_TENANT\_INDEX in table IAU\_COMMON is missing the required columns or index itself is missing. This maybe caused by a known issue, anyway, this missing index will be added in 12.2.2 upgrade.

INFO Audit schema index DYN\_USER\_INDEX in table IAU\_COMMON is missing the required columns or index itself is missing. This maybe caused by a known issue, anyway, this missing index will be added in 12.2.2 upgrade.

INFO Audit schema index DYN\_COMPONENT\_TYPE\_INDEX in table IAU\_COMMON is missing the required columns or index itself is missing. This maybe caused by a known issue, anyway, this missing index will be added in 12.2.2 upgrade.

INFO Audit schema index DYN\_USER\_TENANT\_INDEX in table IAU\_COMMON is missing the required columns or index itself is missing. This maybe caused by a known issue, anyway, this missing index will be added in 12.2.2 upgrade.

Completed index test for table IAU\_COMMON: TEST\_REQUIRED\_INDEXXES --> Test that the table contains all the required indexes +++ FAIL

 **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 12c using the RCU.

## Creating the Required 12c Schemas Using RCU (Optional)

When upgrading from 11g, you must create the required 12c schemas. If your setup has non-SSL ports open, you can use the Upgrade Assistant to create schemas by using the default schema settings. In case of SSL enabled setup, you can use the Repository Creation Utility (RCU) to create customized schemas. This procedure describes how to create schemas using the RCU. Information about using the Upgrade Assistant to create schemas is covered in the upgrade procedures.

 **Note:**

This step is not required for non-SSL setup, as the Upgrade Assistant creates the necessary 12c schemas during the upgrade process.

For SSL enabled setup, you must run the RCU to create the necessary 12c schemas.

 **Note:**

If you are upgrading from a previous 12c release of Oracle Fusion Middleware, you do not need to re-create these schemas if they already exist. Refer to the steps below to identify the existing schemas in your domain.

The following schemas must exist before you upgrade to 12c. If you are upgrading from 11g, and you are not sure which schemas you currently have, refer to the steps below to identify the existing schemas in your domain. You do not need to re-create these schemas if they already exist.

- **Service Table** schema (*prefix\_STB*). This schema is new in 12c and is required for domain-based upgrades. It stores basic schema configuration information (for example, schema prefixes and passwords) that can be accessed and used by other Oracle Fusion Middleware components during the domain creation. This schema is automatically created when you run the Repository Creation Utility (RCU), where you specify the existing schema owner prefix that you used for your other 11g schemas.

 **Note:**

If the Service Table schema does not exist, you may encounter the error message UPGAST-00328 : The schema version registry table does not exist on this database. If that happens it is necessary to create the service table schema in order to run Upgrade Assistant

- **Oracle Platform Security Services (OPSS)** schema (*prefix\_OPSS*). This schema is required if you are using an OID-based security store in 11g. This schema is automatically created when you run the Repository Creation Utility (RCU). The only supported LDAP-based OPSS security store is Oracle Internet Directory (OID). An LDAP-based policy store is typically used in production environments. You do not need to reassociate an OID-based security store before upgrade. While the Upgrade Assistant is running, you can select the OPSS schema. The Upgrade Assistant upgrades the OID-based security store automatically.

 **Note:**

The 12c OPSS database schema is required so that you can reference the 12c schema during the reconfiguration of the domain. Your domain continues to use the OID-based security store after the upgrade is complete.

To create the 12c schemas with the RCU:

1. (Optional) If you are upgrading from 11g, and you wish to confirm the schemas which are present in your existing domain, then connect to the database as a user with DBA privileges, and run the following code 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 ORDER BY MRC_NAME, COMP_ID ;
```

2. Verify that a certified JDK already exists on your system by running `java -version` from the command line. For 12c (12.2.1.3.0), the certified JDK is 1.8.0\_131 and later.

Ensure that the `JAVA_HOME` environment variable is set to the location of the certified JDK. For example:

- (UNIX) `setenv JAVA_HOME=/home/Oracle/Java/jdk1.8.0_131`
- (Windows) `set JAVA_HOME=C:\home\Oracle\Java\jdk1.8.0_131`

Add `$JAVA_HOME/bin` to `$PATH`.

3. Go to the `oracle_common/bin` directory:
  - (UNIX) `NEW_ORACLE_HOME/oracle_common/bin`
  - (Windows) `NEW_ORACLE_HOME\oracle_common\bin`
4. Start the RCU:
  - (UNIX) `./rcu`
  - (Windows) `rcu.bat`
5. On the Welcome screen, click **Next**.
6. On the Create Repository screen, select **Create Repository** and then select **System Load and Product Load**.

If you do not have DBA privileges, select **Prepare Scripts for System Load**. This will generate a SQL script containing all the same SQL statements and blocks that would have been called if the RCU were to execute the actions for the selected components. After the script is generated, a user with the necessary SYS or SYSDBA privileges can execute the script to complete the system load phase. Click **Next**.

7. On the Database Connection Details screen, select the **Database Type** and enter the connection information for the database that hosts the 11g schemas. See the pertinent table below.

**Table 4-4 Connection Credentials for Oracle Databases and Oracle Databases with Edition-Based Redefinition**

Option	Description and Example
Host Name	Specify the name of the server where your database is running in the following format: <code>examplehost.exampledomain.com</code> For Oracle RAC databases, specify the SCAN name or one of the node names in this field.
Port	Specify the port number for your database. The default port number for Oracle databases is 1521.
Service Name	Specify the service name for the database. Typically, the service name is the same as the global database name. For Oracle RAC databases, specify the service name of one of the nodes in this field. For example: <code>examplehost.exampledomain.com</code>
Username	Enter the user name for your database. The default user name is SYS.

**Table 4-4 (Cont.) Connection Credentials for Oracle Databases and Oracle Databases with Edition-Based Redefinition**

Option	Description and Example
Password	Enter the password for your database user.
Role	Select the database user's role from the drop-down list: <b>Normal</b> or <b>SYSDBA</b>

8. On the Select Components screen, select **Select existing prefix** and select the prefix that was used to create the existing 11g schemas from the drop-down menu (for example, `DEV11G`). This prefix is used to logically group schemas together for use in this domain. Select the following schemas:

- If you are upgrading an SSL enabled setup, select the following schemas:
  - User Messaging Service (`prefix_UMS`)
  - Weblogic Services (`prefix_WLS`)
  - Audit services (`prefix_IAU_APPEND` and `prefix_IAU_VIEWER`)

 **Note:**

The Common Infrastructure Services (`prefix_STB`) is selected by default. IAU is greyed out if 11g is configured for Audit Data Store).

- If you are upgrading a non-SSL enabled setup, select the following schemas:
  - Weblogic Services (`prefix_WLS`)
  - Audit services (`prefix_IAU_APPEND` and `prefix_IAU_VIEWER`)

 **Note:**

The Common Infrastructure Services (`prefix_STB`) is selected by default. IAU is greyed out if 11g is configured for Audit Data Store).

 **Note:**

The Common Infrastructure Services (`prefix_STB`) and Oracle Platform Security Services (`prefix_OPSS`) schemas are selected by default if they have not yet been created.

Make a note of the prefix and schema names for the components you are installing as you will need this information when you configure the installation. Click **Next**.

9. In the Checking Prerequisites dialog, verify that the prerequisites check is successful, then click **OK**.

10. On the Schema Passwords screen, specify the passwords for your schema owners.

Make a note of the passwords you enter on this screen as you will need this information while configuring your product installation.

11. On the Map Tablespaces screen, configure the required tablespace mapping for the schemas you want to create.

Click **Next**, then click **OK** in the confirmation dialog. When the progress dialog shows the tablespace creation is complete, click **OK**.

You see the **Encrypt Tablespace** check box only if you have enabled Transparent Data Encryption (TDE) in the database (Oracle or Oracle EBR) when you start the RCU. Select the **Encrypt Tablespace** check box on the Map Tablespaces screen to encrypt all new tablespaces that the RCU creates.

12. Verify the information on the Summary screen and click **Create** to begin schema creation.

This screen contains information about the log files that were created from this RCU operation. Click on the name of a particular log file to view the contents of that file.

13. Review the information on the Completion Summary screen to verify that the operation is completed successfully. Click **Close** to complete the schema creation.

## Stopping Servers and Processes

Before you run the Upgrade Assistant to upgrade your 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.

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.

### Note:

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 System Components

To stop 11g system components, such as Oracle HTTP Server, use the `opmnctl` script:

 **Note:**

If the Oracle HTTP server is shared with other services, then you can choose *not* to stop the Oracle HTTP server.

- (UNIX) `OHS_INSTANCE_HOME/bin/opmnctl stopall`
- (Windows) `OHS_INSTANCE_HOME\bin\opmnctl stopall`

You can stop system components in any order.

### Step 2: 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:

When prompted, enter your user name and password.

**Method 1:** 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 2:** 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')  
  
wls:/offline>nmKill('ManagedServerName')
```

### Step 3: 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 following methods to stop the Administration Server:

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

**Method 1:** To stop a 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 2:** 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')  
  
wls:/offline>nmKill('AdminServer')
```

#### Step 4: Stop Node Manager

To stop Node Manager, run the following command:

```
kill $(ps -ef | grep nodemanager | | awk '{print $2}')
```

## Upgrading Schemas on OIMHOST1

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

### Note:

For SSL enabled setup, it is mandatory to run the Repository Creation Utility (RCU) to upgrade the existing schemas. For more information, see [Creating the Required 12c Schemas Using RCU \(Optional\)](#) For non-SSL enabled setup, running RCU to upgrade schemas is optional.

- [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.

- [Identifying Existing Schemas Available for Upgrade](#)  
This optional task enables you to review the list of available schemas before you begin the upgrade by querying the schema version registry. The registry contains schema information such as version number, component name and ID, date of creation and modification, and custom prefix.
- [Starting the Upgrade Assistant](#)  
Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12c (12.2.1.3.0). Oracle

recommends that you run the Upgrade Assistant as a non-SYSDBA user, completing the upgrade for one domain at a time.

- [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 task enables you to review the list of available schemas before you begin the upgrade by querying the schema version registry. The registry contains schema information such as version number, component name and ID, date of creation and modification, and custom prefix.

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 ORDER BY MRC_NAME, COMP_ID;
```

2. Examine the report that is generated.

If an upgrade is not needed for a schema, the `schema_version_registry` table retains the schema at its pre-upgrade version.

3. Note the schema prefix name that was used for your existing schemas. You will use the same prefix when you create new 12c schemas.

### Notes:

- If you used an OID-based policy store in 11g, 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 12c (12.2.1.3.0). Do not attempt to upgrade a domain that includes components that are not yet available for upgrade to 12c (12.2.1.3.0).

## Starting the Upgrade Assistant

Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12c (12.2.1.3.0). Oracle recommends that you run the Upgrade Assistant as a non-SYSDBA user, completing the upgrade for one domain at a time.

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 ensure that UTF-8 is used by the JVM, use the JVM option -  
`Dfile.encoding=UTF-8`.

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`

### Note:

In the above command, `ORACLE_HOME` refers to the 12c (12.2.1.3.0) Oracle Home.

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

## Upgrading Oracle Identity Manager Schemas Using the Upgrade Assistant

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

### Note:

- If the pre-upgrade environment has Audit schema (IAU), you must first upgrade Audit schema only, using the **Individually Selected Schema** option on the Selected Schemas screen, and selecting **Oracle Audit Services** schema. Ensure that you select the appropriate IAU schema from the list of available IAU schemas. The upgrade assistant will not detect the corresponding IAU schema from the provided domain directory automatically. Hence, you must select it manually. Once the IAU schema is upgraded, run the Upgrade Assistant again to upgrade the remaining schemas using the **All Schema Used by a domain** option on the Selected Schemas screen.
- If there is no Audit schema (IAU) in your pre-upgrade environment, use the **All Schema Used by a Domain** option on the Selected Schemas screen and proceed.
- To check whether the pre-upgrade environment has the IAU schema, run the following SQL command using the user with sysdba privileges:

```
select username from dba_users where username like '%IAU%';
```

This command lists the IAU schemas available in your configured database.

### Note:

For SSL enabled setup, it is mandatory to run the Repository Creation Utility (RCU) to upgrade the existing schemas. For more information, see [Creating the Required 12c Schemas Using RCU \(Optional\)](#). For non-SSL enabled setup, running RCU to upgrade schemas is optional.

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 Selected Schemas 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 12c (12.2.1.3.0) components. Do not upgrade schemas that are currently being used to support components that are not included in Oracle Fusion Middleware 12c (12.2.1.3.0).

- **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 your OIM database has only the SSL port open, 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.

Click **Next**.

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.

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 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.

 **Note:**

The component ID or schema name is changed for UCSUMS schema as of release 12.1.2, which means the Upgrade Assistant does not automatically recognize the possible schemas and display them in a drop-down list. You must manually enter the name in a text field. The name can be either *prefix\_ORASDPM* or *prefix\_UMS*, depending on the starting point for the upgrade.

**11g to 12c Upgrades Only:** The UCSUMS schema is not auto-populated. Enter *prefix\_ORASDPM* as the user. The upgrade environment uses *\_ORASDPM* as the schema name, whereas in the 12c environment it is referred to as *\_UMS*.

6. 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.

7. 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.

8. 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**.

9. If the upgrade is successful: On the Upgrade Success screen, 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:

```
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 ORDER BY MRC_NAME, COMP_ID ;
```

In the query result:

- 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 12.2.1.3.0.

### Note:

However, that not all schema versions will be updated. Following are the schemas that do not require an upgrade to this release and will retain their pre-upgrade version number:

```
BIPLATFORM
IAU
IAU_APPEND
IAU_VIEWER
OPSS
UCSUMS
WLS
```

- 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 on OIMHOST1

Run the Reconfiguration Wizard on OIMHOST1 to reconfigure your domain component configurations to 12c (12.2.1.3.0).

- [About Reconfiguring the Domain](#)  
Run the Reconfiguration Wizard to reconfigure your domain component configurations to 12c (12.2.1.3.0).

### About Reconfiguring the Domain

Run the Reconfiguration Wizard to reconfigure your domain component configurations to 12c (12.2.1.3.0).

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:

- 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.

- If the installation that you're upgrading does not use Oracle Access Manager (OAM), then you must edit two files to prevent the Reconfiguration Wizard from attempting to update the nonexistent OAM Infrastructure schema, which causes the upgrade to fail.

Comment out the lines in your `$DOMAIN_HOME/init-info/domain-info.xml` that are similar to this example:

Where, `DOMAIN_HOME` is the Administrator server domain home.

```
<!--extention-template-ref name="Oracle Identity Navigator"
  version="11.1.1.3.0"
  location="/u01/app/oracle/product/fmw/iam111130/common/
templates/applications/oracle.oinav_11.1.1.3.0_template.jar"
symbol="" /-->
<!--install-comp-ref name="oracle.idm.oinav"
version="11.1.1.3.0"

symbol="oracle.idm.oinav_11.1.1.3.0_iam111130_ORACLE_HOME"
  product_home="/u01/app/oracle/product/fmw/iam111130" /-->
```

and similarly comment out the lines in `$DOMAIN_NAME/config/config.xml` that are similar to this example:

```
<!--app-deployment>
  <name>oinav#11.1.1.3.0</name>
  <target>AdminServer</target>
  <module-type>ear</module-type>

  <source-path>/u01/app/oracle/product/fmw/iam111130/oinav/
modules/oinav.ear_11.1.1.3.0/oinav.ear</source-path>
  <deployment-order>500</deployment-order>
  <security-dd-model>DDOnly</security-dd-model>
  <staging-mode>nostage</staging-mode>
</app-deployment-->
```

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.

Follow these instructions to reconfigure the existing domain using the Reconfiguration Wizard. See Reconfiguring WebLogic Domains in *Upgrading Oracle WebLogic Server*.

- [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.

To create a backup of the Administration server domain directory:

1. Copy the source domain to a separate location to preserve the contents.  
(Windows) `copy /Oracle/Middleware/user_projects/domains to /Oracle/Middleware/user_projects/domains_backup.`  
(UNIX) `cp -rf mydomain mydomain_backup`
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](#).

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 environment variable `WLS_ALTERNATIVE_TYPES_DIR` using 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.

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:

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

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

So, first, change the cache directory by setting the environment variable `CONFIG_JVM_ARGS`.

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

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.

### Note:

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 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 12c (12.2.1.3.0) is 1.8.0\_131 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 11g datasource, the reconfiguration will preserve the existing values. For new datasources where the schema was created for 12c 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 11g, and your database has `_OPSS` or `_IAU` 11g database schemas, you must manually enter database connection details for those schemas. These schemas were not required in 11g 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.

6. 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**.
7. 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.
8. 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**.

9. 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.

10. 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**.

11. 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 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 OIMHOST1 only.

- [Upgrading Domain Component Configurations](#)  
After reconfiguring the domain, use the Upgrade Assistant to upgrade the domain *component* configurations inside the domain to match the updated domain configuration.

## Upgrading Domain Component Configurations

After reconfiguring the domain, 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 12c (12.2.1.3.0). Oracle recommends that you run the Upgrade Assistant as a non-SYSDBA user, completing the upgrade for one domain at a time.
- [Upgrading Oracle Identity Manager Domain Component Configurations](#)  
Navigate through the screens in the Upgrade Assistant to upgrade component configurations in the WebLogic domain.
- [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 12.2.1.3.0.

## Starting the Upgrade Assistant

Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12c (12.2.1.3.0). Oracle recommends that you run the Upgrade Assistant as a non-SYSDBA user, completing the upgrade for one domain at a time.

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 ensure that UTF-8 is used by the JVM, use the JVM option -  
`Dfile.encoding=UTF-8`.

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`

 **Note:**

In the above command, `ORACLE_HOME` refers to the 12c (12.2.1.3.0) Oracle Home.

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
<code>-readiness</code>	Required for readiness checks <b>Note:</b> 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 <code>-examine</code> parameter.

Table 4-5 (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 <i>silent mode</i> (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 <i>attribute</i>	Optional	Sets the logging level, specifying one of the following attributes: <ul style="list-style-type: none"> <li>• TRACE</li> <li>• NOTIFICATION</li> <li>• WARNING</li> <li>• ERROR</li> <li>• INCIDENT_ERROR</li> </ul> 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 4-5 (Cont.) Upgrade Assistant Command-Line Parameters**

Parameter	Required or Optional	Description
-logDir <i>location</i>	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) <code>ORACLE_HOME/oracle_common/upgrade/logs</code> <code>ORACLE_HOME/oracle_common/upgrade/temp</code> (Windows) <code>ORACLE_HOME\oracle_common\upgrade\logs</code> <code>ORACLE_HOME\oracle_common\upgrade\temp</code>
-help	Optional	Displays all of the command-line options.

## Upgrading Oracle Identity Manager Domain Component Configurations

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

After running the Reconfiguration Wizard to reconfigure the WebLogic domain to 12c (12.2.1.3.0), you must 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, enter the WebLogic domain directory path.

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. If there are remote managed servers hosting User Messaging Services (UMS) configuration files: On the UMS Configuration screen, provide the credentials to these servers so that the Upgrade Assistant can access the configuration files.

 **Note:**

You may need to manually copy the UMS configuration files if the Upgrade Assistant is unable to locate them. See [Error while Copying User Messaging Service \(UMS\) Configuration Files](#).

6. On the Old (that is, 11g) OIM Home Location screen, select **11g Source**, and specify the absolute path to the 11.1.2.3.0 OIM Oracle Home, which is `ORACLE_HOME/Oracle_IDM`.

Click **Next**.

7. 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.

8. 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.

9. 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**.

10. 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 `NEW_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 12.2.1.3.0.

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

To sign in to the Administration Console in an EDG deployment, see [Validating the Virtual Server Configuration and Access to the Consoles](#).

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

 **Note:**

- After upgrade, ensure you run the administration tools from the new 12c 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.
- In the site-specific configuration, the WebLogic and EM consoles must be accessible with the URLs either directly or through proxy URLs.

## Starting the Servers

After you upgrade Oracle Identity Manager, start the servers.

You must start the servers in the following order:

1. Start the Administration Server.

If Node manager is configured, do not start the Node Manager.

2. From the terminal, start the Oracle SOA Suite managed server with the Administration Server URL, and the BPM property set to `TRUE`.

For example:

```
./startManagedWebLogic.sh <SOA_Managed_server> t3://
weblogic_admin_host:weblogic_admin_port -Dbpm.enabled=true
```

 **Note:**

- For first boot, Oracle SOA Suite managed server must be started manually by using the command in the above example.
- In SSL environment, when starting managed servers for the first time for bootstrap, provide the non-SSL port number of the Administration Server.

3. Once the SOA server is in running state, from the terminal, start the Oracle Identity Manager Managed Server with the Administration Server URL.

This time, OIM bootstrap process will be executed, and after successful bootstrap, OIM Managed Server will be shut down automatically.  
For example:

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

 **Note:**

As in the previous step, provide the non-SSL port number of the Administration Server.

4. Shut down the SOA Managed Server and the Administration Server. For information about stopping the servers and processes, see [Stopping Servers and Processes](#).

## Replicating the Domain Configurations on each OIMHOST

Replicate the domain configurations on OIMHOST2. This involves packing the upgraded domain on OIMHOST1 and unpacking it 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 on OIMHOST2.



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.

## Copying `oracle.iam.ui.custom-dev-starter-pack.war` to the 12c (12.2.1.3.0) Oracle Home

After you run the Upgrade Assistant for config upgrade, manually copy the `oracle.iam.ui.custom-dev-starter-pack.war` file from backup of 11g Oracle home to 12c (12.2.1.3.0) Oracle home.

Complete the following steps:

1. From the 11g release folder, go to location: `ORACLE_HOME/Oracle_IDM1/server/apps/`
2. Copy the file: `oracle.iam.ui.custom-dev-starter-pack.war`
3. For 12c (12.2.1.3.0) release, go to location: `ORACLE_HOME/idm/server/apps/` and paste the copied `.war` file.

## Starting the Servers on OIMHOST1 and OIMHOST2

After you upgrade Oracle Identity Manager on both OIMHOST1 and OIMHOST2, start the servers.

You must start the servers in the following order:

1. Start the Node Manager on both OIMHOST1 and OIMHOST2.
2. Start the Administration Server on OIMHOST1.
3. Start the Oracle SOA Suite Managed Server (without BPM property) and Oracle Identity Manager Managed Servers on OIMHOST1.
4. Start the Oracle SOA Suite Managed Server (without BPM property) and Oracle Identity Manager Managed Servers on OIMHOST2.

- [Starting Servers and Processes](#)  
After a successful upgrade, start all processes and servers, including the Administration Server and any Managed Servers.
- [Configuring Oracle HTTP Servers to Front End OIM, and SOA Managed Servers](#)

## 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 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

Starting the Node Manager in the Administration Server domain home:

- (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.

**Method 1:** To start a Administration Server, run the following command:

```
nohup DOMAIN_HOME/bin/startWeblogic.sh &
```

**Method 2:** To start a Administration Server by using node manager, run the following commands:

```
cd ORACLE_COMMON_HOME/common/bin
./wlst.sh
wlst offline> nmConnect('nodemanager_username','nodemanager_password',
                        'ADMINVHN','5556','domain_name',
```

```

                                'DOMAIN_HOME')
nmStart('AdminServer')

```

### Step 3: Start the Managed Servers

#### Note:

In an HA environment, it is preferred to use the console or node manager to start servers.

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**.

## Configuring Oracle HTTP Servers to Front End OIM, and SOA Managed Servers

Complete the following steps:

1. On each of the web servers on WEBHOST1 and WEBHOST2, create a file named `mod_wls_ohs.conf` in the directory `OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/instances/OHS_INSTANCE_NAME`.

This file must contain the following information:

```

# oim admin console(idmshell based)
<Location /admin>
  SetHandler weblogic-handler
  WLCookieName oimjsessionid
  WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
  WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

# oim self and advanced admin webapp consoles(canonic webapp)

<Location /oim>
  SetHandler weblogic-handler
  WLCookieName oimjsessionid
  WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
  WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

<Location /identity>
  SetHandler weblogic-handler
  WLCookieName oimjsessionid
  WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
  WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"

```

```
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>

  <Location /sysadmin>
    SetHandler weblogic-handler
    WLCookieName oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>

# SOA Callback webservice for SOD - Provide the SOA Managed Server Ports
  <Location /sodcheck>
    SetHandler weblogic-handler
    WLCookieName oimjsessionid
    WebLogicCluster soavhn1.example.com:7003,soavhn2.example.com:7003
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>

# Callback webservice for SOA. SOA calls this when a request is approved/
rejected
# Provide the OIM Managed Server Port
  <Location /workflowservice>
    SetHandler weblogic-handler
    WLCookieName oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>

# xlWebApp - Legacy 9.x webapp (struts based)
  <Location /xlWebApp>
    SetHandler weblogic-handler
    WLCookieName oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>

# Nexaweb WebApp - used for workflow designer and DM
  <Location /Nexaweb>
    SetHandler weblogic-handler
    WLCookieName oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>

# used for FA Callback service.
  <Location /callbackResponseService>
    SetHandler weblogic-handler
    WLCookieName oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
```

```

        WLProxySSL ON
        WLProxySSLPassThrough ON
    </Location>

# spml xsd profile
<Location /spml-xsd>
    SetHandler weblogic-handler
    WLCookieName    oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>

<Location /HTTPClnt>
    SetHandler weblogic-handler
    WLCookieName    oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>

<Location /reqsvc>
    SetHandler weblogic-handler
    WLCookieName oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>

<Location /integration>
    SetHandler weblogic-handler
    WLCookieName oimjsessionid
    WebLogicCluster soavhn1.example.com:7003,soavhn2.example.com:7003
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>

<Location /provisioning-callback>
    SetHandler weblogic-handler
    WLCookieName oimjsessionid
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>

<Location /CertificationCallbackService>
    SetHandler weblogic-handler
    WLCookieName JSESSIONID
    WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>

```

```
<Location /ucs>
  SetHandler weblogic-handler
  WLCookieName oimjsessionid
  WebLogicCluster soavhnl.example.com:7003,soavhn2.example.com:7003
  WLLogFile /tmp/web_log.log
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

<Location /FacadeWebApp>
  SetHandler weblogic-handler
  WLCookieName oimjsessionid
  WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
  WLLogFile /tmp/web_log.log
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

<Location /iam/governance/configmgmt>
  SetHandler weblogic-handler
  WLCookieName oimjsessionid
  WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
  WLLogFile /tmp/web_log.log
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

<Location /iam/governance/scim/v1>
  SetHandler weblogic-handler
  WLCookieName oimjsessionid
  WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
  WLLogFile /tmp/web_log.log
  WLProxySSL ON
  WLProxySSLPassThrough ON>
</Location>

<Location /iam/governance/token/api/v1>
  SetHandler weblogic-handler
  WLCookieName oimjsessionid
  WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
  WLLogFile /tmp/web_log.log
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

<Location /OIGUI>
  SetHandler weblogic-handler
  WLCookieName oimjsessionid
  WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
  WLLogFile /tmp/web_log.log
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

<Location /iam/governance/applicationmanagement>
  SetHandler weblogic-handler
  WLCookieName oimjsessionid
  WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
  WLLogFile /tmp/web_log.log
  WLProxySSL ON
  WLProxySSLPassThrough ON
```

```

</Location>

<Location /iam/governance/adminservice/api/v1>
  SetHandler weblogic-handler
  WLCookieName oimjsessionid
  WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
  WLLogFile /tmp/web_log.log
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

<Location /iam/governance/selfservice/api/v1>
  SetHandler weblogic-handler
  WLCookieName oimjsessionid
  WebLogicCluster oimvhn1.example.com:14000,oimvhn2.example.com:14000
  WLLogFile /tmp/web_log.log
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

```

2. Save the file on both WEBHOST1 and WEBHOST2.
3. Stop and start the Oracle HTTP Server instances on both WEBHOST1 and WEBHOST2.
4. 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.

## 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, replace the 12c (12.2.1.3.0) `ORACLE_HOME/idm/designconsole/config/xlconfig.xml` file with the 11.1.2.3.0 `ORACLE_HOME/Oracle_IDM1/designconsole/config/xlconfig.xml` file.

## Completing the Post-Upgrade Tasks for SSL Enabled Setup

If you are upgrading an Oracle Identity Manager SSL enabled setup, you must perform the required post-upgrade tasks to complete the upgrade process.

Complete the following tasks if you have upgraded an SSL enabled setup:

1. Changes done for SSL settings in `setDomainEnv.sh`, `startWeblogic.sh`, `startManagedWeblogic.sh`, and `datasources` are lost after upgrade. Re-do all of the changes.
2. Start the WebLogic Administration Server. To start the Administration Server, use the `startWebLogic` script:
  - (UNIX) `DOMAIN_HOME/bin/startWebLogic.sh`
  - (Windows) `DOMAIN_HOME\bin\startWebLogic.cmd`

Where, `DOMAIN_HOME` is the Administration domain.

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

3. Make necessary changes to the following newly created datasources, for SSL settings:

- LocalSvcTblDataSource
- opss-audit-DBDS
- opss-audit-viewDS
- opss-data-source
- WLSSchemaDataSource

For information about updating the newly created datasources, see *Updating Datasource oimOperationsDB Configuration in Administering Oracle Identity Governance*

4. In case of Customer Identity and Java Standard Trust, import your identity trust certificate to the new JDK home. The 12c (12.2.1.3.0) uses `jdk1.8.0_131`. To import the identity trust certificate to the new JDK home, use the following command:

```
./keytool -importcert -alias startssl -keystore JAVA_HOME/jre/lib/security/cacerts -storepass changeit -file supportcert.pem
```

5. Verify that all of the SSL configuration changes including the SSL port related changes done in 11g (pre upgrade), are present post upgrade. If the changes are lost, you must redo them post upgrade. Some of the SSL configuration changes include:

- OimFrontEndURL
- backOfficeURL
- SOA Server URL
- ForeignJNDIProvider-SOA

For more information about configuring SSL for Oracle Identity Governance, see *Updating Oracle Identity Governance in Administering Oracle Identity Governance*.

## Installing Standalone Oracle BI Publisher

When you upgrade Oracle Identity Manager 11.1.2.3.0 to Oracle Identity Governance 12c (12.2.1.3.0), the embedded Oracle BI Publisher present in the 11.1.2.3.0 deployment, is removed. Therefore, you must install a new standalone Oracle BI Publisher 12c (12.2.1.3.0) post upgrade, for configuring the Oracle Identity Governance reports.

For information about installing and configuring Oracle BI Publisher 12c (12.2.1.3.0), see *Installing and Configuring Oracle BI Publisher in Developing and Customizing Applications for Oracle Identity Governance*.

For information about integrating standalone Oracle BI Publisher with Oracle Identity Governance 12c (12.2.1.3.0), see *Integrating Standalone BI Publisher with Oracle Identity Governance in Developing and Customizing Applications for Oracle Identity Governance*.



# 5

## Upgrading OIM-OAM Integrated Environments set up Manually

You can upgrade Oracle Identity Manager (OIM), Oracle Access Manager (OAM) integrated split domain highly available environments that are set up manually, from 11g Release 2 (11.1.2.3.0) to 12c (12.2.1.3.0) using the upgrade procedure described in this section.

### Note:

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

### Topics

- [About the OIM-OAM Integrated HA Topology Set Up Manually](#)  
The sample topology is based on the split domain four node topology described in the *Enterprise Deployment Guide for Oracle Identity and Access Management 11g Release 2* (11.1.2.3.0), that is deployed manually.
- [Supported Starting Points for Integrated HA Upgrade](#)  
Review the supported starting points for each of the components in your integrated environment in order to upgrade to 12c (12.2.1.3.0). If the components are in earlier versions, upgrade them to the version that is supported for 12c upgrade.
- [Roadmap for Upgrading OIM-OAM Integrated Highly Available Environments Set Up Manually](#)  
Refer to the roadmap for upgrading Oracle Identity Manager and Oracle Access Manager integrated highly available 11.1.2.3.0 environments that was set up manually, to 12c (12.2.1.3.0).
- [Enabling Oracle Identity Governance 12c to Work With Oracle Access Manager 11g](#)  
If Oracle Access Manager is still in 11.1.2.3.0 and if Oracle Identity Manager is upgraded to 12c (12.2.1.3.0), the OIM-OAM integration breaks down due to the mismatch in JDK version.

## About the OIM-OAM Integrated HA Topology Set Up Manually

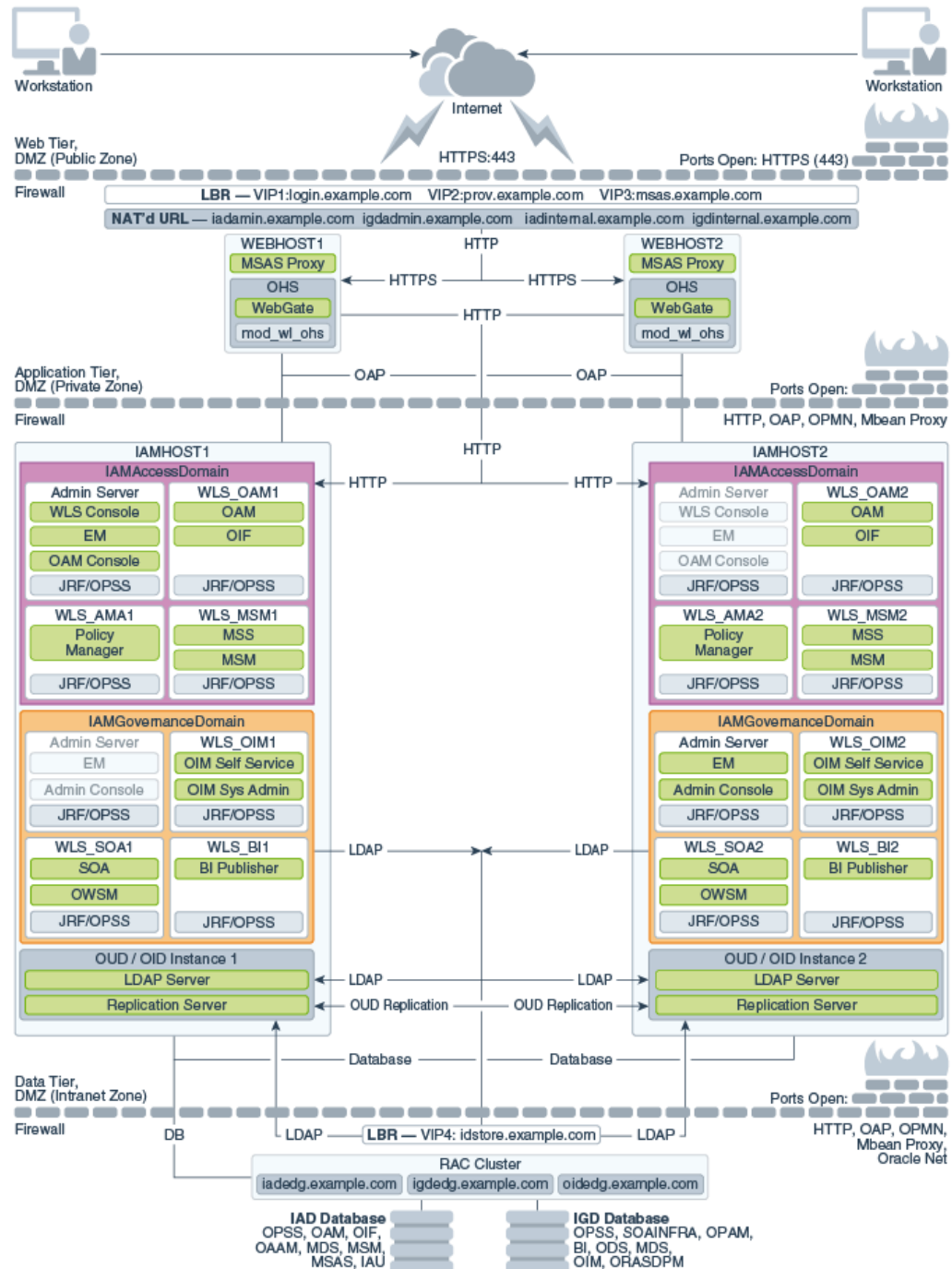
The sample topology is based on the split domain four node topology described in the *Enterprise Deployment Guide for Oracle Identity and Access Management 11g Release 2* (11.1.2.3.0), that is deployed manually.

See [Enterprise Deployment Guide for Oracle Identity and Access Management](#).

This topology and the accompanying procedures in this chapter are provided to serve as an example for upgrading a highly available, integrated Oracle Identity and Access Management environment. Your specific Oracle Identity and Access Management installation will vary, but this topology and upgrade procedure demonstrates the key elements of the upgrade process, which can be applied to your specific environment.

For a complete description of the topology diagram, refer to the [Enterprise Deployment Guide for Oracle Identity and Access Management](#) in the 11g Release 2 (11.1.2.3.0) Documentation Library.

**Figure 5-1 OIM-OAM Integrated Topology Set Up Manually**



## Supported Starting Points for Integrated HA Upgrade

Review the supported starting points for each of the components in your integrated environment in order to upgrade to 12c (12.2.1.3.0). If the components are in earlier versions, upgrade them to the version that is supported for 12c upgrade.

The following table lists the versions that are supported for upgrade of an integrated highly available environments.

**Table 5-1 Supported Starting Point for Integrated HA Upgrade**

Component	Supported Starting Point
Oracle Identity Manager	11g Release 2 (11.1.2.3.0)
Oracle Access Manager	11g Release 2 (11.1.2.3.0)
Oracle Adaptive Access Manager	11g Release 2 (11.1.2.3.0)
Oracle SOA Suite	11g Release 1 (11.1.1.9.0)
Oracle WebLogic Server	10.3.6

Oracle Adaptive Access Manager is not part of the Oracle Identity and Access Management suite for 12c (12.2.1.3.0), and hence will not be upgraded to 12c. Oracle Adaptive Access Manager 11.1.2.3.0 is compatible with Oracle Access Manager 12c (12.2.1.3.0).

## Roadmap for Upgrading OIM-OAM Integrated Highly Available Environments Set Up Manually

Refer to the roadmap for upgrading Oracle Identity Manager and Oracle Access Manager integrated highly available 11.1.2.3.0 environments that was set up manually, to 12c (12.2.1.3.0).

The following table describes the tasks that you must perform to upgrade an OIM-OAM integrated topology described in [About the OIM-OAM Integrated HA Topology Set Up Manually](#).

**Table 5-2 Tasks for Upgrading Integrated Environments Set Up Manually**

Task	Documentation
Review the OIM-OAM integrated topology.	See <a href="#">About the OIM-OAM Integrated HA Topology Set Up Manually</a> .
Review the supported starting points for integrated environment upgrade.	See, <a href="#">Supported Starting Points for Integrated HA Upgrade</a> .
Ensure that the LDAP server and the Oracle Access Manager have the same lockout value configured before you start the upgrade. That is, the lockout threshold of libOVD, OAM, and LDAP should be the same, else the lock and unlock use cases fail after upgrade.	See <a href="#">Setting the LockoutThreshold in Active Directory</a> in the <i>Oracle Fusion Middleware Deployment Guide for Oracle Identity and Access Management</i> for 11g Release 2 (11.1.2.3.0).
This is applicable for a OIM-OAM integrated single node setup as well.	

**Table 5-2 (Cont.) Tasks for Upgrading Integrated Environments Set Up Manually**

Task	Documentation
If you have configured Node Manager, ensure that the Node Manager is stopped before you proceed with the upgrade.	See <a href="#">Stopping Servers and Processes</a> .
Check if Oracle Access Manager (OAM) is integrated with Oracle Identity Manager (OIM) in a single domain	See <a href="#">Checking if OAM and OAAM is in the Same Domain in an OAM-OAAM-OIM Integrated Setup</a> .
If Oracle Access Manager is integrated with Oracle Identity Manager (OIM), and if both the products are in a same domain, a separate OAM domain needs to be cloned that works with OIM in the source domain. It is the cloned OAM domain that needs to be upgraded to 12c.	
Upgrade Oracle Access Manager to 12c (12.2.1.3.0). You can choose to upgrade Oracle Identity Manager first too.	See <a href="#">Upgrading Oracle Access Manager Highly Available Environments</a> .
Upgrade Oracle Identity Manager to 12c (12.2.1.3.0).	See <a href="#">Upgrading Oracle Identity Manager Highly Available Environments</a> .
If Oracle Access Manager is still in 11g, and if Oracle Identity Manager is upgraded to 12c, you must delete the MD5 entries from the <code>java.security</code> file.	See <a href="#">Enabling Oracle Identity Governance 12c to Work With Oracle Access Manager 11g</a> .

**Note:**

If you encounter any issues during the upgrade, see [Troubleshooting the Oracle Identity Manager Upgrade](#).

## Enabling Oracle Identity Governance 12c to Work With Oracle Access Manager 11g

If Oracle Access Manager is still in 11.1.2.3.0 and if Oracle Identity Manager is upgraded to 12c (12.2.1.3.0), the OIM-OAM integration breaks down due to the mismatch in JDK version.

To resolve this, on the 12c (12.2.1.3.0) OIM server, you must edit the `jdk1.8.0_131/jre/lib/security/java.security` file to remove the MD5 related entries by doing the following:

1. Remove the MD5 entry from the following line:

```
jdk.certpath.disabledAlgorithms=MD2, MD5, RSA keySize < 1024, \
  DSA keySize < 1024, EC keySize < 224
```

After removing the MD5 entry, the line should look like the following:

```
jdk.certpath.disabledAlgorithms=MD2, RSA keySize < 1024, \
    DSA keySize < 1024, EC keySize < 224
```

2. Remove the MD5 entry from the following line:

```
jdk.tls.disabledAlgorithms=SSLv3, RC4, MD5withRSA, DH keySize <
768, \    EC keySize < 224
```

After removing the MD5 entry, the line should look like the following:

```
jdk.tls.disabledAlgorithms=SSLv3, RC4, DH keySize < 768, \    EC
keySize < 224
```

3. Remove the MD5 entry from the following line:

```
jdk.jar.disabledAlgorithms=MD2, MD5, RSA keySize < 1024
```

After removing the MD5 entry, the line should look like the following:

```
jdk.jar.disabledAlgorithms=MD2, RSA keySize < 1024
```

4. Remove the MD5 entries from the following block:

```
jdk.xml.dsig.secureValidationPolicy=\
    disallowAlg http://www.w3.org/TR/1999/REC-xslt-19991116,\
    disallowAlg http://www.w3.org/2001/04/xmldsig-more#rsa-md5,\
    disallowAlg http://www.w3.org/2001/04/xmldsig-more#hmac-md5,\
    disallowAlg http://www.w3.org/2001/04/xmldsig-more#md5,\
    maxTransforms 5,\
    maxReferences 30,\
    disallowReferenceUriSchemes file http https,\
    minKeySize RSA 1024,\
    minKeySize DSA 1024,\
    noDuplicateIds,\
    noRetrievalMethodLoops
```

After removing the MD5 entries, the block should look like the following:

```
jdk.xml.dsig.secureValidationPolicy=\
    disallowAlg http://www.w3.org/TR/1999/REC-xslt-19991116,\
    maxTransforms 5,\
    maxReferences 30,\
    disallowReferenceUriSchemes file http https,\
    minKeySize RSA 1024,\
    minKeySize DSA 1024,\
    noDuplicateIds,\
    noRetrievalMethodLoops
```

5. Reboot the OIM domain.



**Note:**

After you upgrade Oracle Access Manager to 12c (12.2.1.3.0), undo the changes you made in this procedure.

# 6

## Upgrading OIM-OAM Integrated Environments set up Using Life Cycle Management Tool

If you had set up an Oracle Identity Manager – Oracle Access Manager integrated environment in 11g Release 2 (11.1.2.3.0) using the Life Cycle Management (LCM) tool, follow the instructions in this chapter to upgrade the same to 12c (12.2.1.3.0).

### Note:

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

### Topics

- [About the OIM-OAM Integrated HA Topology Set Up Using LCM Tool](#)  
The sample topology is based on the split domain eight node topology described in the *Enterprise Deployment Guide for Oracle Identity and Access Management 11g Release 2 (11.1.2.3.0)*, that is deployed using the Life Cycle Management (LCM) tool.
- [Supported Starting Points](#)  
Review the supported starting points for each of the components in your integrated environment in order to upgrade to 12c (12.2.1.3.0). If the components are in earlier versions, upgrade them to the version that is supported for 12c upgrade.
- [Roadmap for Upgrading OIM-OAM Integrated Environments set up Using Life Cycle Management Tool](#)  
Refer to the roadmap in this section for upgrading Oracle Identity Manager and Oracle Access Manager integrated highly available 11.1.2.3.0 environments, set up using Life Cycle Management (LCM) tool, to 12c (12.2.1.3.0).

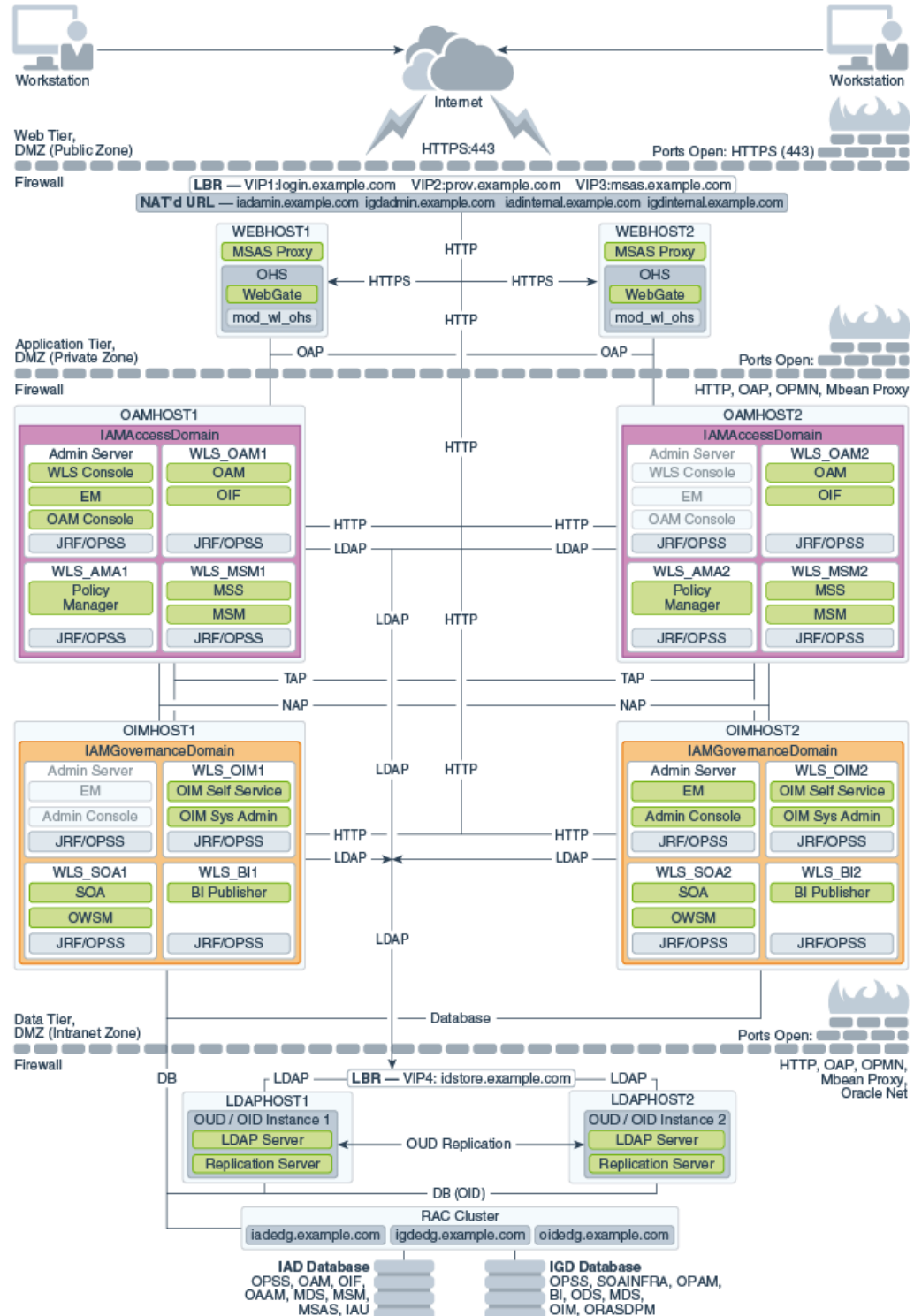
## About the OIM-OAM Integrated HA Topology Set Up Using LCM Tool

The sample topology is based on the split domain eight node topology described in the *Enterprise Deployment Guide for Oracle Identity and Access Management 11g Release 2 (11.1.2.3.0)*, that is deployed using the Life Cycle Management (LCM) tool.

This topology and the accompanying procedures in this chapter are provided to serve as an example for upgrading a highly available, integrated Oracle Identity and Access Management environment. Your specific Oracle Identity and Access Management installation will vary, but this topology and upgrade procedure demonstrates the key elements of the upgrade process, which can be applied to your specific environment.

For a complete description of the topology diagram, refer to the *Enterprise Deployment Guide in the Oracle Identity and Access Management in the 11g Release 2 (11.1.2.3.0) Documentation Library*.

**Figure 6-1 OIM-OAM Integrated Topology Set Up Using LCM Tool**





## Supported Starting Points

Review the supported starting points for each of the components in your integrated environment in order to upgrade to 12c (12.2.1.3.0). If the components are in earlier versions, upgrade them to the version that is supported for 12c upgrade.

The following table lists the versions that are supported for upgrade of an integrated highly available environments.

**Table 6-1 Supported Starting Points for Integrated HA Upgrade**

Component	Supported Starting Point
Oracle Identity Manager	11g Release 2 (11.1.2.3.0)
Oracle Access Manager	11g Release 2 (11.1.2.3.0)
Oracle Adaptive Access Manager	11g Release 2 (11.1.2.3.0)
Oracle SOA Suite	11g Release 1 (11.1.1.9.0)
Oracle WebLogic Server	10.3.6

Oracle Adaptive Access Manager is not part of the Oracle Identity and Access Management suite for 12c (12.2.1.3.0), and hence will not be upgraded to 12c. Oracle Adaptive Access Manager 11.1.2.3.0 is compatible with Oracle Access Manager 12c (12.2.1.3.0).

## Roadmap for Upgrading OIM-OAM Integrated Environments set up Using Life Cycle Management Tool

Refer to the roadmap in this section for upgrading Oracle Identity Manager and Oracle Access Manager integrated highly available 11.1.2.3.0 environments, set up using Life Cycle Management (LCM) tool, to 12c (12.2.1.3.0).

The following table describes the tasks that you must perform to upgrade an integrated topology described in [About the OIM-OAM Integrated HA Topology Set Up Using LCM Tool](#).

**Table 6-2 Tasks for Upgrading Integrated Environments Set Up Using LCM Tool**

Task	Documentation
Review the OIM-OAM integrated topology.	See <a href="#">About the OIM-OAM Integrated HA Topology Set Up Using LCM Tool</a> .
Review the supported starting points for integrated environment upgrade.	See <a href="#">Supported Starting Points</a> .
Ensure that the LDAP server and the Oracle Access Manager have the same lockout value configured before you start the upgrade. That is, the lockout threshold of libOVD, OAM, and LDAP should be the same, else the lock and unlock use cases fail after upgrade.	See <a href="#">Setting the LockoutThreshold in Active Directory</a> in the <i>Oracle Fusion Middleware Deployment Guide for Oracle Identity and Access Management</i> for 11g Release 2 (11.1.2.3.0).
This is applicable for a OIM-OAM integrated single node setup as well.	

**Table 6-2 (Cont.) Tasks for Upgrading Integrated Environments Set Up Using LCM Tool**


Task	Documentation
If you have configured Node Manager, ensure that the Node Manager is stopped before you proceed with the upgrade.	See <a href="#">Stopping Servers and Processes</a> .
Check if Oracle Access Manager (OAM) is integrated with Oracle Identity Manager (OIM) in a single domain	See <a href="#">Checking if OAM and OAAM is in the Same Domain in an OAM-OAAM-OIM Integrated Setup</a> .
If Oracle Access Manager is integrated with Oracle Identity Manager (OIM), and if both the products are in a same domain, a separate OAM domain needs to be cloned that works with OIM in the source domain. It is the cloned OAM domain that needs to be upgraded to 12c.	
Upgrade the Oracle Identity Manager on OIMHOST1 shared domain to 12c (12.2.1.3.0).	See <a href="#">Upgrading Oracle Identity Manager Single Node Environments</a> .
Do NOT start the servers after you upgrade.	
Take a backup and delete the contents of the private domain.	See <a href="#">soa-infra Application is in 'Prepared' State Post Upgrade</a> .
It is recommended that you perform this step, or the <code>soa-infra</code> application continues to be in <code>Prepared</code> state instead of <code>active</code> state, post upgrade.	
Pack the Oracle Identity Manager shared domain and unpack it into the private domain on OIMHOST1 and OIMHOST2.	See <a href="#">Replicating the Domain Configurations on each OIMHOST</a> .

 **Note:**

Use the pack and unpack commands as described in the above section to pack the OIM shared domain and unpack it to the private domain on OIMHOST1 and OIMHOST2.

**Table 6-2 (Cont.) Tasks for Upgrading Integrated Environments Set Up Using LCM Tool**

Task	Documentation
<p>Start the Oracle SOA Suite Managed Servers and Oracle Identity Manager servers on OIMHOST1.</p> <p>You must start the Administration Server from the shared domain and the Managed Servers from the private domain.</p> <p>When you start the Oracle SOA Suite Managed Servers for the first time after upgrade, ensure that you do it with Business Process Management (BPM) property and Administration Server URL.</p>	<p>See <a href="#">Starting the Servers</a>.</p> <p>If bootstrapping fails when you start the Oracle Identity Manager servers for the first time, follow the instructions described in <a href="#">OIM Bootstrap for DEPLOYSOACOMPOSITES Task Fails After Upgrade</a> to resolve this, and then start the servers,</p>
<p>Restart the Administration Server, Oracle SOA Suite Managed Servers, and the Oracle Identity Manager Managed Servers on OIMHOST1 and start the Oracle SOA Suite Managed Servers, and the Oracle Identity Manager Managed Servers on OIMHOST2.</p> <p>When you restart the Oracle SOA Suite Managed Servers for the second time after upgrade, ensure that you do it without Business Process Management (BPM) property.</p>	<p>See <a href="#">Stopping Servers and Processes</a> for stopping the servers.</p> <p>See <a href="#">Starting the Servers</a> for starting the servers.</p>
<p>Upgrade the Oracle Access Manager on OAMHOST1 shared domain to 12c (12.2.1.3.0).</p> <p>Do not start the servers after you upgrade.</p>	<p>See <a href="#">Upgrading Oracle Access Manager Single Node Environments</a>.</p>
<p>Pack the Oracle Access Manager shared domain and unpack it to the private domain on OAMHOST1 and OAMHOST2.</p>	<p>See <a href="#">Replicating the Domain Configurations on each OAMHOST</a>.</p>
<p>Start the Administration Server and the Oracle Access Manager Managed Servers.</p>	<p>See <a href="#">Starting Servers and Processes</a> .</p>

 **Note:**

Use the pack and unpack commands as described in the above section to pack the OAM shared domain and unpack it to the private domain on OAMHOST1 and OAMHOST2.

 **Note:**

If you encounter any issues when upgrading Oracle Identity Manager , see [Troubleshooting the Oracle Identity Manager Upgrade](#) for troubleshooting tips.

# 7

## Troubleshooting the Oracle Identity Manager Upgrade

If you encounter errors while upgrading Oracle Identity Manager, review the following troubleshooting procedures.

### Note:

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

- [WebLogic Server is Not in the Running Status](#)  
After upgrading, the WebLogic admin server does not change to the running status.
- [Upgrading Product Schemas Error: OIMR2PS2\\_OIM.PK\\_POP](#)  
When you are upgrading the product schemas, the unique constraint OIMR2PS2\_OIM.PK\_POP violated error is displayed.
- [Upgrading Product Schemas Error: OIMR2PS2\\_OIM.UK\\_ENTITY\\_TYPE](#)  
When you are upgrading the product schemas, the OIMR2PS2\_OIM.UK\_ENTITY\_TYPE violated error is displayed.
- [KeystoreService Exception in the Logs After Reconfiguring the OIM Domain](#)  
After you reconfigure the Oracle Identity Manager (OIM) domain, the logs show some exceptions which can be ignored.
- [Warning when Generating the Pre-Upgrade Report for OIM](#)  
When you run the pre-upgrade report utility to generate the pre-upgrade report for Oracle Identity Manager, the audit store instantiation failure warning is seen on the console, which can be ignored.
- [Domain Reconfiguration Error](#)  
When using the reconfiguration utility you might notice a *Domain reconfig* error. To fix this error, you need to manually delete the related applications and references.
- [OIM Bootstrap for DEPLOYSOACOMPOSITES Task Fails After Upgrade](#)  
After you complete the Oracle Identity Manager upgrade, when you start the Oracle Identity Manager Managed Servers for the first time, bootstrapping happens. If the OIM bootstrap fails for DEPLOYSOACOMPOSITES task, use the workaround described in this section to resolve the issue.
- [Authorization Policy Merge Issue](#)
- [MAR Update or Metadata Merge Issue](#)  
When you start the Oracle Identity Manager Managed Servers for the first time after upgrade, if you encounter any error during the bootstrap process which is related to MARUPDATE bootstrap task, run the external utility

`mergeMDSDataAfterUpgrade.sh` from the 12c Middleware Home to re-trigger the Metadata Services (MDS) merge process.

- [Error When Opening ADF DI Excel Sheet After Upgrade](#)  
The ADFDI functionality will not work after you upgrade Oracle Identity Manager to 12c (12.2.1.3.0).
- [Compilation Error When Starting the SOA Server After Upgrade](#)  
When you start the Oracle SOA Suite for the first time after upgrade, you may see the compilation error in the SOA server logs.
- [Warning in Oracle Identity Manager Server Logs After Upgrade](#)  
After upgrade, the Oracle Identity Manager (OIM) Server logs show NPE warning, which can be ignored.
- [Default Challenges Questions are not Updated After Upgrade](#)  
After you upgrade Oracle Identity Manager 11.1.2.3.0 to 12c, the default challenge questions are not updated. It still shows the old or existing challenge questions.
- [OPSS Processing Error When Reconfiguring the Domain](#)  
When you upgrade a Oracle Identity Manager in an integrated environment, the OPSS processing error is encountered.
- [EditFailedException When Releasing Configuration From WebLogic Console](#)  
After you upgrade Oracle Identity Manager to 12c (12.2.1.3.0), when you click **Release Configuration** on Oracle WebLogic Console, the following error is seen:
- [OIM Application Deployment Fails Intermittently](#)  
After you upgrade Oracle Identity Manager to 12c (12.2.1.3.0), the `oim` application deployment may fail with the following error:
- [soa-infra Application is in 'Prepared' State Post Upgrade](#)  
After you upgrade Oracle Identity Manager (OIM) and Oracle Access Manager (OAM) integrated environment that was set up using Life Cycle Management (LCM) tool, the `soa-infra` application continues to be in `Prepared` mode, instead of showing `active` mode.
- [Oracle Identity Manager Server Throws OutOfMemoryError](#)  
When you start the servers post upgrade, `OutOfMemoryError` is thrown.
- [SOA Fails to Join Coherence Cluster During the First Start After Upgrade](#)  
After you upgrade Oracle Identity Manager (OIM) and Oracle Access Manager (OAM) integrated environment, when you start the Oracle SOA Suite Server for the first time, the coherence cluster fails to start with the following error:
- [LDAP User Create and Update Reconciliation Job Fails](#)  
LDAP User Create and Update Reconciliation job fails to run with the following exception:
- [BI Managed Server is Seen on WebLogic Console After Upgrade](#)  
If your 11g Release 2 (11.1.2.3.0) domain had a custom name for BI Managed Server, that is, the name other than `bi_server1`, then this Managed Server will not be deleted during the upgrade.
- [Empty Pages or Panels After Upgrade](#)  
After you complete the upgrade, the Applications tile in the OIM Self Service console and the Import/Export links in the Admin console may be rendered as empty pages or panels.

## WebLogic Server is Not in the Running Status

After upgrading, the WebLogic admin server does not change to the running status.

After upgrading to Weblogic Server to 12c (12.2.1.3.0), the admin server on WebLogic Server 12c (12.2.1.3) does not start. It does not display any relevant message for the current state in the admin server logs.

While tacking a thread dump we see:

```
"[STANDBY] ExecuteThread: '3' for queue: 'weblogic.kernel.Default (self-tuning)'" #39 daemon prio=5 os_prio=0 tid=<tid#> nid=0x1c38a runnable [0x00007fc4348bc000]
java.lang.Thread.State: RUNNABLE
at java.net.SocketInputStream.socketRead0(Native Method)
at java.net.SocketInputStream.socketRead(SocketInputStream.java:116)
at java.net.SocketInputStream.read(SocketInputStream.java:170)
at java.net.SocketInputStream.read(SocketInputStream.java:141)
at weblogic.nodemanager.common.NMReader.readLength(NMReader.java:223)
at weblogic.nodemanager.common.NMReader.readChunk(NMReader.java:200)
at weblogic.nodemanager.common.NMReader.readLine(NMReader.java:41)
at
weblogic.nodemanager.common.NMInputOutput.readLine(NMInputOutput.java:30)
)
at
weblogic.nodemanager.common.DataFormat.readResponse(DataFormat.java:504)
at
weblogic.nodemanager.client.NMServerClient.getResponseString(NMServerClient.java:900)
at
weblogic.nodemanager.client.NMServerClient.checkResponse(NMServerClient.java:884)
at
weblogic.nodemanager.client.NMServerClient.connect(NMServerClient.java:733)
at
weblogic.nodemanager.client.NMServerClient.checkConnected(NMServerClient.java:668)
at
weblogic.nodemanager.client.NMServerClient.checkConnected(NMServerClient.java:674)
at
weblogic.nodemanager.client.NMServerClient.initState(NMServerClient.java:519)
- locked <0x00000000f58683a8> (a
weblogic.nodemanager.client.PlainClient)
at
weblogic.nodemanager.mbean.NodeManagerRuntime.initState(NodeManagerRuntime.java:600)
at
weblogic.server.ServerLifecycleRuntime.<init>(ServerLifecycleRuntime.java:161)
at
weblogic.server.ServerLifecycleService.createServerLifecycleRuntime(Serv
```

```

erLifecycleService.java:263)
at
weblogic.server.ServerLifecycleService.start(ServerLifecycleService.java
:149)
at
weblogic.server.AbstractServerService.postConstruct(AbstractServerServic
e.java:76)
at sun.reflect.GeneratedMethodAccessor5.invoke(Unknown Source)
at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessor
Impl.java:43)
at java.lang.reflect.Method.invoke(Method.java:497)
at
org.glassfish.hk2.utilities.reflection.ReflectionHelper.invoke(Reflectio
nHelper.java:1287)
at
org.jvnet.hk2.internal.ClazzCreator.postConstructMe(ClazzCreator.java:33
3)
at org.jvnet.hk2.internal.ClazzCreator.create(ClazzCreator.java:375)
at
org.jvnet.hk2.internal.SystemDescriptor.create(SystemDescriptor.java:487
)
at
org.glassfish.hk2.runlevel.internal.AsyncRunLevelContext.findOrCreate(As
yncRunLevelContext.java:305)
at
org.glassfish.hk2.runlevel.RunLevelContext.findOrCreate(RunLevelContext.
java:85)
at org.jvnet.hk2.internal.Utilities.createService(Utilities.java:2126)
at
org.jvnet.hk2.internal.ServiceHandleImpl.getService(ServiceHandleImpl.ja
va:116)
- locked <0x00000000e228e260> (a java.lang.Object)
at
org.jvnet.hk2.internal.ServiceHandleImpl.getService(ServiceHandleImpl.ja
va:90)
at
org.glassfish.hk2.runlevel.internal.CurrentTaskFuture$QueueRunner.oneJob
b(CurrentTaskFuture.java:1237)
at
org.glassfish.hk2.runlevel.internal.CurrentTaskFuture$QueueRunner.run(Cu
rrentTaskFuture.java:1168)
at
weblogic.work.SelfTuningWorkManagerImpl$WorkAdapterImpl.run(SelfTuningWo
rkManagerImpl.java:678)
at
weblogic.invocation.ComponentInvocationContextManager._runAs(ComponentInv
ocationContextManager.java:352)
at
weblogic.invocation.ComponentInvocationContextManager.runAs(ComponentInv
ocationContextManager.java:337)
at
weblogic.work.LivePartitionUtility.doRunWorkUnderContext(LivePartitionUt
ility.java:57)
at

```



```

weblogic.work.PartitionUtility.runWorkUnderContext(PartitionUtility.java
:41)
at
weblogic.work.SelfTuningWorkManagerImpl.runWorkUnderContext(SelfTuningWo
rkManagerImpl.java:652)
at weblogic.work.ExecuteThread.execute(ExecuteThread.java:420)
at weblogic.work.ExecuteThread.run(ExecuteThread.java:360)

```

The admin server is connected to a node manager that is not running correctly or replying correctly.

As per stack trace we can see that communication to the node manager is established. By checking `NMServerClient.java` source, we see that it is still reading, but not progressing. This might be due to a network issue. In thread dumps, we can see that this is an on going process.

the same error may occur if the admin server is connecting to a node manager running on a different version of the WebLogic Server. In a clustered setup, this might occur if the node manager is running on any of the node in the cluster.

To solve this issue:

Ensure that the node manager on all nodes in the cluster setup is shut down. This allows the WLS admin Server to start-up without any issue.

Also, ensure that the node manager in the remote machine is of the same WebLogic Server version. In case of Weblogic upgrade, ensure that the node manager is not running on any of the nodes in the cluster.

For more information, see [Doc ID 2431508.1](#).

## Upgrading Product Schemas Error: OIMR2PS2\_OIM.PK\_POP

When you are upgrading the product schemas, the unique constraint OIMR2PS2\_OIM.PK\_POP violated error is displayed.

The following error message is displayed:

```

2020-04-24T02:48:32.875-07:00] [OIM] [INCIDENT_ERROR] []
[upgrade.OIM.OIM1]
[tid: 90] [ecid: ab5f4fa6-a864-4c03-b79e-464f0b408557-00000002,0] [[
oracle.iam.oimupgrade.exceptions.OIMUpgradeException: SQL Exception in
running Upgrade Scripts
    at
oracle.iam.oimupgrade.onehop.SchemaUpgradeManager.upgrade(SchemaUpgradeM
anager
.java:314)
    at oracle.iam.oimupgrade.mrua.OIMMRUA.upgrade(OIMMRUA.java:203)
    at oracle.ias.update.plugin.Plugin.upgrade(Plugin.java:730)
    at oracle.ias.update.plan.PlanStep.upgrade(PlanStep.java:736)
    at
oracle.ias.update.PhaseProcessor$UpgradeProcessor.runStepPhase(PhaseProc
essor.
java:726)
    at oracle.ias.update.PhaseProcessor.runStep(PhaseProcessor.java:369)
    at

```

```

oracle.ias.update.PhaseProcessor$ExtendedRunnable.run(PhaseProcessor.java:1058)
)
    at
java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142)
)
    at
java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:617)
)
    at java.lang.Thread.run(Thread.java:748)
Caused by: java.sql.SQLException: ORA-20100: Exception encountered!!
ORA-00001: unique constraint (OIMR2PS2_OIM.PK_POP) violated

ORA-06512: at line 268
ORA-06512: at line 468

```

To solve this issue, complete the following steps:

1. From the SQL developer, connect to OIM schema.
2. Open **POP** OIM DB table details.
3. Go to the **index** tab.
4. Manually delete the **PK\_POP** index.

## Upgrading Product Schemas Error: OIMR2PS2\_OIM.UK\_ENTITY\_TYPE

When you are upgrading the product schemas, the OIMR2PS2\_OIM.UK\_ENTITY\_TYPE violated error is displayed.

The following error message is displayed:

```

ORA-00001: unique constraint (OIMR2PS2_OIM.UK_ENTITY_TYPE) violated
ORA-06512: at line 23

```

To solve the error, see [Doc ID 2412393.1](#).

## KeystoreService Exception in the Logs After Reconfiguring the OIM Domain

After you reconfigure the Oracle Identity Manager (OIM) domain, the logs show some exceptions which can be ignored.

The following exceptions are seen in the logs after you reconfigure the OIM domain:

```

oracle.security.jps.upgrade.tools.KeyStoreUpgrade - Exception in
checking for
jdk cacert store
oracle.security.jps.service.keystore.KeyStoreServiceException: Failed
to load
the keystore.

```

```

at
oracle.security.jps.internal.keystore.ldap.KeyStoreDataManager.getKeyStore(KeyStoreDataManager.java:987)
at
oracle.security.jps.internal.keystore.ldap.LdapKeyStoreServiceImpl.getKeyStore(LdapKeyStoreServiceImpl.java:279)
at
oracle.security.jps.upgrade.tools.KeyStoreUpgrade.importJdkCacerts(KeyStoreUpgrade.java:313)
at
oracle.security.jps.upgrade.tools.KeyStoreUpgrade.upgradeDITAndData(KeyStoreUpgrade.java:266)
at
oracle.security.jps.upgrade.tools.utility.Upgrade.upgradeOPSSDITAndData(Upgrade.java:1078)
at
oracle.security.jps.upgrade.tools.utility.Upgrade.upgradeOPSS(Upgrade.java:772)
at
oracle.security.opss.tools.lifecycle.OpssDomainConfigImpl.reconfigSubsystem(OpssDomainConfigImpl.java:359)
at
oracle.security.opss.tools.lifecycle.OpssDomainConfigImpl.initializeSubsystem(OpssDomainConfigImpl.java:271)
at
oracle.security.opss.tools.lifecycle.cie.OpssSecurityConfiguration.initializeSubsystem(OpssSecurityConfiguration.java:188)
at
com.oracle.cie.domain.progress.template.importer.ImporterOPSSProcessingPhase.initialize(ImporterOPSSProcessingPhase.java:36)
at
com.oracle.cie.domain.progress.domain.generation.OPSSProcessingPhase.processOPSS(OPSSProcessingPhase.java:154)
at
com.oracle.cie.domain.progress.domain.generation.OPSSProcessingPhase.execute(OPSSProcessingPhase.java:54)
at
com.oracle.cie.domain.progress.AbstractProgressGenerator.run(AbstractProgressGenerator.java:94)
at java.lang.Thread.run(Thread.java:745)
Caused by: java.io.IOException: Keystore publiccacerts in app stripe system does not exist

```

```

at
oracle.security.jps.internal.keystore.provider.FarmKeyStoreSpi.engineLoad(Farm
KeyStoreSpi.java:606)
at java.security.KeyStore.load(KeyStore.java:1479)
at
oracle.security.jps.internal.keystore.ldap.KeyStoreDataManager.getKeyStore(Key
StoreDataManager.java:976)

```

Ignore this warning and proceed.

## Warning when Generating the Pre-Upgrade Report for OIM

When you run the pre-upgrade report utility to generate the pre-upgrade report for Oracle Identity Manager, the audit store instantiation failure warning is seen on the console, which can be ignored.

The following warning is seen on the console when generating the pre-upgrade report for OIM:

```

WARNING: Audit store instantiation failure, type: db reason:
java.lang.ClassNotFoundException:
oracle.security.audit.config.dynamic.persistence.internal.ldap.LdapAudit
Store
Jul 28, 2016 10:26:05 PM
oracle.security.jps.az.internal.runtime.service.PDPServiceImpl
oracle.security.jps.az.internal.runtime.service.PDPServiceImpl
SEVERE: Cannot read the default policy store.
oracle.security.jps.service.policystore.PolicyStoreException:
oracle.security.jps.az.internal.management.pd.PD
at
oracle.security.jps.az.common.pd.service.PDServiceFinder.getPolicyDistri
bution
Service(PDServiceFinder.java:65)
at
oracle.security.jps.az.internal.runtime.service.PDPServiceImpl.initializ
eMixed
Mode(PDPServiceImpl.java:714)
at
oracle.security.jps.az.internal.runtime.service.PDPServiceImpl.initial(P
DPServ
iceImpl.java:685)

```

Ignore this warning and proceed.

## Domain Reconfiguration Error

When using the reconfiguration utility you might notice a *Domain reconfig* error. To fix this error, you need to manually delete the related applications and references.

During upgrade from 11g to 12c (12.2.1.3.0), the upgrade wizard displays the following error in the domain reconfiguration step:

```
Domain reconfig failed with:
CFGFWK-40951: Correct source path of the applications to refer to the
new
installation.
2017-11-09 03:53:00,804 SEVERE [42]
com.oracle.cie.domain.progress.AbstractProgressGenerator - Error
occurred in
phase {Validating Domain} execution.
com.oracle.cie.domain.ValidateException: CFGFWK-60953: An application or
library was not relocated to the new MW home.
CFGFWK-60953: An error in the reconfiguration templates resulted in the
following deployments being left in the original MW home:
OIMMetadata#11.1.1.3.0
oim#11.1.1.3.0
oracle.webcenter.skin#11.1.1@11.1.1
oracle.soa.bpel#11.1.1@11.1.1
oracle.soa.worklist#11.1.1@11.1.1
oracle.soa.b2b#11.1.1@11.1.1

CFGFWK-60953: Correct the reconfiguration templates
    at
com.oracle.cie.domain.progress.domain.reconfig.wlscore.ValidateDomainPha
se.val

    idateDeployments(ValidateDomainPhase.java:124)
        at
com.oracle.cie.domain.progress.domain.reconfig.wlscore.ValidateDomainPha
se.exe

    cute(ValidateDomainPhase.java:48)
        at
com.oracle.cie.domain.progress.AbstractProgressGenerator.run(AbstractPro
gressG

    enerator.java:94)
        at java.lang.Thread.run(Thread.java:748)
```

To fix the *Domain reconfig* error, complete the following steps:

1. Open to files `DOMAIN_HOME/config/config.xml` and `DOMAIN_HOME/init-info/config-groups.xml`.
2. Delete the following applications and references:
  - `oracle.soa.bpel#11.1.1@11.1.1`
  - `oracle.soa.worklist#11.1.1@11.1.1`
  - `oracle.soa.b2b#11.1.1@11.1.1`
  - `oracle.webcenter.skin#11.1.1@11.1.1`
3. Save and close the files.

## OIM Bootstrap for DEPLOYSOACOMPOSITES Task Fails After Upgrade

After you complete the Oracle Identity Manager upgrade, when you start the Oracle Identity Manager Managed Servers for the first time, bootstrapping happens. If the OIM bootstrap fails for DEPLOYSOACOMPOSITES task, use the workaround described in this section to resolve the issue.

The following error is seen in the OIM server logs:

```
<Oct 4, 2016, 4:53:51,904 AM PDT> <Info>
<oracle.iam.OIMPostConfigManager>
<BEA-000000> <FROM THREAD:Processing sar=/ORACLE_HOME/idm/server/
workflows/composites/scajars/sca_DefaultRequest Approval_rev5.0.jar>
<Oct 4, 2016, 4:53:51,906 AM PDT> <Info>
<oracle.iam.OIMPostConfigManager>
<BEA-000000> <FROM THREAD:Adding sar file -/ORACLE_HOME/idm/server/
workflows/composites/scajars/sca_DefaultRequestAppro val_rev5.0.jar>
<Oct 4, 2016, 4:53:52,40 AM PDT> <Info>
<oracle.iam.OIMPostConfigManager>
<BEA-000000> <FROM THREAD:INFO: Creating HTTP connection to
host:slc09pgg, port:16230>
<Oct 4, 2016, 4:53:54,694 AM PDT> <Info>
<oracle.iam.OIMPostConfigManager>
<BEA-000000> <FROM THREAD:INFO: Received HTTP response from the server,
response code=500>
<Oct 4, 2016, 4:53:54,695 AM PDT> <Info>
<oracle.iam.OIMPostConfigManager>
<BEA-000000> <FROM THREAD:---->Response code=500, error:There was an
error deploying the composite on soa_server1: keepInstancesOnRedeploy
flag can only be used with BPM enabled installation..>
<Oct 4, 2016, 4:53:54,696 AM PDT> <Info>
<oracle.iam.OIMPostConfigManager>
<BEA-000000> <FROM THREAD:> <Oct 4, 2016, 4:53:54,964 AM PDT> <Info>
<oracle.iam.OIMPostConfigManager>
<BEA-000000> <Completed the script Command execution.>
<Oct 4, 2016, 4:53:54,965 AM PDT> <Info>
<oracle.iam.OIMPostConfigManager>
<BEA-000000> <The logs are written to file :/tmp/
deploySOAComposites_1475582008428.log>
<Oct 4, 2016, 4:53:54,966 AM PDT> <Info>
<oracle.iam.OIMPostConfigManager>
<BEA-000000> < [OIM_CONFIG] Error while executing the wlst script /tmp/
deploySOAComposites_1475582008428.py>
<Oct 4, 2016, 4:53:54,967 AM PDT> <Error>
<oracle.iam.OIMPostConfigManager>
<BEA-000000> < Error while executing the wlst script /tmp/
deploySOAComposites_1475582008428.py>
<Oct 4, 2016, 4:53:54,967 AM PDT> <Error>
<oracle.iam.OIMPostConfigManager>
<BEA-000000> < Error while executing the wlst script /tmp/
deploySOAComposites_1475582008428.py>
```

```
<Oct 4, 2016, 4:53:54,967 AM PDT> <Info>
<oracle.iam.OIMPostConfigManager>
<BEA-000000> < deploySOAComposites() Failed.>
<Oct 4, 2016, 4:53:54,968 AM PDT> <Info>
<oracle.iam.OIMPostConfigManager>
<BEA-000000> < Forced deployment of 12c SOA composite failed.>
<Oct 4, 2016, 4:53:54,968 AM PDT> <Warning>
<oracle.iam.OIMPostConfigManager>
<BEA-000000> < Unable to deploy te SOA Composites.>
<Oct 4, 2016, 4:53:54,968 AM PDT> <Warning>
<oracle.iam.OIMPostConfigManager>
<BEA-000000> < Unable to deploy te SOA Composites.>
<Oct 4, 2016, 4:53:54,969 AM PDT> <Info>
<oracle.iam.OIMPostConfigManager>
<BEA-000000> <Reason of fail :Error occurred while deploying the 12c
SOA composite>
```

The following error is seen in the Oracle SOA Suite (SOA) server logs:

```
<Oct 4, 2016, 2:57:30,535 AM PDT> <Error> <ServletContext-/soa-infra>
<BEA-000000> <Error during deployment
oracle.fabric.common.FabricDeploymentException: keepInstancesOnRedeploy
flag
can only be used with BPM enabled installation. {rootCauses=[]}
at
oracle.integration.platform.blocks.deploy.servlet.DeployProcessor.doDepl
oyWork
(DeployProcessor.java:582)
at
oracle.integration.platform.blocks.deploy.servlet.DeployProcessor.doDepl
oyWork
(DeployProcessor.java:473)
at
oracle.integration.platform.blocks.deploy.servlet.DeployProcessor.doDepl
oy(Dep
loyProcessor.java:282)
at
oracle.integration.platform.blocks.deploy.servlet.DeployProcessor.proces
s(Depl
oyProcessor.java:168)
at
oracle.integration.platform.blocks.deploy.servlet.CompositeDeployerServl
et.doP
ostInsideLoggingSession(CompositeDeployerServlet.java:250)
Truncated. see log file for complete stacktrace
<Oct 4, 2016, 2:57:30,553 AM PDT> <Error>
<oracle.integration.platform.blocks.deploy.servlet> <SOA-21537>
<Sending back
error message: There was an error deploying the composite on
soa_server1:
keepInstancesOnRedeploy flag can only be used with BPM enabled
installation...>
```

To resolve this issue, start the Oracle SOA Suite server with the following property:

```
-Dbpm.enabled=true
```

This completes the OIM bootstrap tasks successfully. After the successful completion of OIM bootstrap tasks, restart all of the servers. This time, do not use the property `-Dbpm.enabled=true` for starting the SOA server. When you start the Managed Servers for the first time after upgrade, start them with the Administration Server URL.

## Authorization Policy Merge Issue

Oracle Identity Manager 11.1.2.3.0 has two Oracle Platform Security Services (OPSS) application policy stripes namely `oim` and `OracleIdentityManager`, whereas Oracle Identity Governance 12.2.1.3 has only one OPSS application policy stripe named `oim`. The 12c upgrade process handles the merging of application stripes into one along with all the customization, at various phases.

If you encounter any error or issue related to OPSS application policies after upgrade, or if you find the policies in inconsistent state, complete the following steps to restore the OPSS application policies:

1. The Authorization policy backup for OIM lying in OPSS schema is taken by the 12c pre-upgrade utility. This backup folder is located at `oim.outputreportfolder/Auth-Policy-Backup`.

`oim.outputreportfolder` is the name of the pre-upgrade report output folder specified by you in the `preupgrade_report_input.properties` file when you ran the pre-upgrade utility.

The backup folder contains the following files:

- `oim.outputreportfolder/Auth-Policy-Backup/oim.xml` — This is for `oim` application policy stripe of 11.1.2.3.0.
- `oim.outputreportfolder/Auth-Policy-Backup/OracleIdentityManager.xml` — This is for `OracleIdentityManager` application policy stripe of 11.1.2.3.0.

Restore these stripes data in OIM database using the following WLST offline commands:

- ```
migrateSecurityStore(type="appPolicies",
srcApp="OracleIdentityManager ", configFile="DOMAIN_HOME/
config/fmwconfig/jps-config_temp.xml", src="desContextOracle",
dst="migrateStripe",overWrite="true")
```
- ```
migrateSecurityStore(type="appPolicies", srcApp="oim",
configFile="DOMAIN_HOME/config/fmwconfig/jps-config_temp.xml",
src="desContextOIM", dst="migrateStripe",overWrite="true")
```

In the above commands, `DOMAIN_HOME/config/fmwconfig/jps-config_temp.xml` file is a copy of the `DOMAIN_HOME/config/fmwconfig/jps-config.xml` file. The following service instances and JPS contexts are added in this file:

```
<serviceInstance name="serviceInsOracle"
provider="policystore.xml.provider"
location="<oim.outputreportfolder>/Auth-Policy-Backup/
OracleIdentityManager.xml"/> <serviceInstance
name="serviceInsOIM" provider="policystore.xml.provider"
location="<oim.outputreportfolder>/Auth-Policy-Backup/oim.xml"/>
```



```

<jpsContext name="desContextOracle">
<serviceInstanceRef ref="serviceInsOracle"/>
</jpsContext>
<jpsContext name="desContextOIM">
<serviceInstanceRef ref="serviceInsOIM"/>
</jpsContext>
<jpsContext name="migrateStripe">
<serviceInstanceRef ref="policystore.db"/>
</jpsContext>

```

2. Migrate the OracleIdentityManager stripe to oim stripe using the following WLST offline command:

```

migrateSecurityStore(type="appPolicies",
srcApp="OracleIdentityManager", dstApp="oim", configFile=DOMAIN_HOME/
config/fmwconfig/jps-config_temp.xml, src="migrateStripe",
dst="migrateStripe", overwrite="false")

```

3. Merge the 12c Out Of The Box application policies on OIM 11.1.2.3.0 application policy stripe named as oim by doing the following:

- a. Unzip the `ORACLE_HOME/idm/common/templates/wls/oracle.OIM.reconfig.template_1_2.2.1.2.0.jar` file to any temporary location. This temporary location is referred to as `unzip_location`.
- b. Verify that the file `unzip_location/security/authorization/jazn-data.xml` exists.
- c. Run the following WLST offline command:

```

migrateSecurityStore(type="appPolicies", srcApp="oim",
configFile=DOMAIN_HOME/config/fmwconfig/jps-config_temp.xml,
src="12c_context", dst="migrateStripe", overwrite="false")

```

The following service instances and JPS contexts are added in the `DOMAIN_HOME/config/fmwconfig/jps-config_temp.xml` file:

```

<serviceInstance name="serviceIns12c_context"
provider="policystore.xml.provide"
location="unzip_location/security/authorization/jazn-dara.xml"/>
<jpsContext name="12c_context">
<serviceInstanceRef ref="service12c_context"/>
</jpsContext>

```

- d. Delete the OracleIdentityManager stripe using the following WLST command:

```

deleteAppPolicies(appStripe="OracleIdentityManager")

```

## MAR Update or Metadata Merge Issue

When you start the Oracle Identity Manager Managed Servers for the first time after upgrade, if you encounter any error during the bootstrap process which is related to `MARUPDATE` bootstrap task, run the external utility `mergeMDSDataAfterUpgrade.sh` from the 12c Middleware Home to re-trigger the Metadata Services (MDS) merge process.

The upgrade utility merges the existing 11.1.2.3.0 MDS data with 12c Out of the Box (OOTB) to preserve the customization.

When you start the OIM Managed Server for the first time, if you encounter errors for `MARUPDATE` bootstrap task, check if the issue is occurring during the MDS merge process. If so, run an external utility to re-trigger the MDS merge process as described in this section.

To check if the issue is occurring during the MDS merge process, do the following:

1. Connect to the Oracle Identity Manager database.
2. Use the following SQL query to check the status of the `MARUPDATE` bootstrap task:

```
select State from OIMBootState where FEATURENAME='MARUPDATE';
```

3. If the query returns `VALID` or `COMPLETE`, the issue is not because of the MDS merge failure. Therefore, no action is required. If the query returns any other result, run the merge utility to re-trigger the MDS merge process.

To re-trigger the MDS merge process using the merge utility, complete the following steps:

1. The OIM pre-upgrade reports folder must exist on the same machine from which the MDS merge utility is going to be triggered. If the pre-upgrade reports are on a different machine, copy them to the machine from which you wish to run the merge utility. The pre-upgrade report utility takes a backup of the MDS data and saves it in the pre-upgrade reports folder.

The MDS backup data is located at `<oim.outputreportfolder>/MDS-Backup` folder. `<oim.outputreportfolder>` is the path that you specified for the property `oim.outputreportfolder` in the `preupgrade_report_input.properties` file, when generating the pre-upgrade reports for OIM.

2. Run the following command from the location `ORACLE_HOME/idm/server/bin/mergeMDSDataAfterUpgrade.sh`

You must specify the location of the OIM pre-upgrade reports folder. The MDS merge utility that you triggered merges the MDS backup data from the pre-upgrade reports folder with the 12c data OOTB.

3. After the successful completion of the MDS merge process, connect to the OIM database and run the following query: `update OIMBootState set State='COMPLETE' where FEATURENAME='MARUPDATE';`

4. Restart the OIM Managed Server.

(Optional) Enter the result of the procedure here.

## Error When Opening ADF DI Excel Sheet After Upgrade

The ADFDI functionality will not work after you upgrade Oracle Identity Manager to 12c (12.2.1.3.0).

After upgrade, when you open the ADF DI spreadsheet in Excel, the following error is displayed:

```
ADFDI-05587: The client and server versions do not match. Using this
version of the client may result in unexpected behavior or errors.
The client version is 11.1.1.7.0 (6882) but the server at http://
host.example.com:22925/identity/adfdiRemoteServlet expects version
12.2.1.3.0 (16546) using precision 3.
```

To resolve this, uninstall and reinstall the ADF DI Excel plug-in, and then re-download the Excel.

## Compilation Error When Starting the SOA Server After Upgrade

When you start the Oracle SOA Suite for the first time after upgrade, you may see the compilation error in the SOA server logs.

The following error is displayed in the SOA server logs:

```
[2016-07-01T02:04:18.239-07:00] [soa_server1] [ERROR] []
[oracle.soa.bpel.system] [tid: DaemonWorkThread: '8' of WorkManager:
'wm/SOAWorkManager'] [userId: ] [ecid:
4f969dd2-853a-4ddf-be01-0ac2ca0d2210-00000009,0:11854] [APP: soa-infra]
[partition-name: DOMAIN] [tenant-name: GLOBAL] Error while loading
process.[[
The process domain is encountering the following errors while loading
the
process "ApprovalProcess" (composite
"default/DefaultRequestApproval!5.0*soa_c9c16746-016e-40c4-
aaaa-6ccd2d685cb4")
.
: BPEL 1.1 compilation failed.
This error contained an exception thrown
by the underlying process loader
module.
Check the exception trace in the
log (with logging level set to debug mode).
If there is a patch installed on
the server, verify that the bpelcClasspath
domain property includes the patch classes.
```

Check the SOA composites status from Oracle Enterprise Manager console after successful start of the Oracle Identity Manager Managed Server.

If the Enterprise Manager console shows `DefaultRequestApproval!5.0` composite status as actively deployed, ignore this one time error.

If you have upgraded your 11g Release 2 (11.1.2.2.0) environments to 11g Release 2 (11.1.2.3.0), and then to 12c (12.2.1.3.0), you will see the compilation error for `DefaultRequestApproval!3.0` composite. This composite was in use in 11g Release 2 (11.1.2.2.0). Before you upgraded to 11.1.2.3.0, this composite processed all of the inflight requests. After upgrading to 11.1.2.3.0, all of the new requests go via `DefaultRequestApproval!5.0` composite.

`DefaultRequestApproval!3.0` is irrelevant when upgrading from 11.1.2.3.0 to 12c (12.2.1.3.0). Therefore, this compilation error can be ignored.

## Warning in Oracle Identity Manager Server Logs After Upgrade

After upgrade, the Oracle Identity Manager (OIM) Server logs show NPE warning, which can be ignored.

After you upgrade Oracle Identity Manager , the following warning is seen in the OIM Server logs for once:

```
<Warning> <oracle.iam.platform.entitymgr.impl> <BEA-000000>
<EntityManagerConfigImpl.getEntityConfig()..Can throw NPE with
providerType:
RDBMSChildDataProviderProvider Definition: type:
RDBMSChildDataProvider className:
oracle.iam.platform.entitymgr.provider.rdbms.RDBMSChildDataProvider
m_params:
parent_id_column : name:parent_id_column type:string required:true
multiValued:falseid_sequence : name:id_sequence type:string
required:false multiValued:false
table : name:table type:string required:true multiValued:false
data_level_column : name:data_level_column type:string required:false
multiValued:false modify_timestamp_column :
name:modify_timestamp_column type:stringrequired:false
multiValued:false
id_column : name:id_column type:string required:true multiValued:false
optimistic_locking : name:optimistic_locking type:boolean
required:true
multiValued:false
paramName: id_type>
<Apr 18, 2017 9:52:54,122 AM PDT> <Warning>
<oracle.iam.platform.entitymgr.impl> <IAM-0040000> <Cannot load entity
definition - java.lang.NullPointerException at
oracle.iam.platform.entitymgr.impl.EntityManagerConfigImpl.getEntityConf
ig(Ent
ityManagerConfigImpl.java:1164) at
oracle.iam.platform.entitymgr.impl.EntityManagerConfigImpl.getEntityConf
ig(Ent
ityManagerConfigImpl.java:1242)
```

This warning can be ignored.

## Default Challenges Questions are not Updated After Upgrade

After you upgrade Oracle Identity Manager 11.1.2.3.0 to 12c, the default challenge questions are not updated. It still shows the old or existing challenge questions.

If you are using default password policy with default challenge questions, you must modify them manually post upgrade per your organization needs to have a better security.

## OPSS Processing Error When Reconfiguring the Domain

When you upgrade a Oracle Identity Manager in an integrated environment, the OPSS processing error is encountered.

The following exception is seen when you run `reconfig.sh` command to reconfigure the Oracle Identity Manager domain:

```
SEVERE [93] com.oracle.cie.domain.progress.AbstractProgressGenerator -  
Error occurred in  
phase {OPSS Processing} execution.  
java.lang.IllegalStateException: SecurityContext: Domain Name:  
IAMGovernanceDomain  
JDBC URL: opss-audit-DBDS:jdbc:oracle:thin:@//slc03rmj:1521/IDMDB  
JDBC URL: opss-data-source:jdbc:oracle:thin:@//slc03rmj:1521/idmdb  
le.com  
Caused by: java.security.InvalidKeyException: Illegal key size  
at javax.crypto.Cipher.checkCryptoPerm(Cipher.java:1039)  
at javax.crypto.Cipher.implInit(Cipher.java:805)  
at javax.crypto.Cipher.chooseProvider(Cipher.java:864)  
at javax.crypto.Cipher.init(Cipher.java:1396)  
at javax.crypto.Cipher.init(Cipher.java:1327)
```

To resolve this issue, do the following:

1. Install the Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files from the following location:  
<http://www.oracle.com/technetwork/java/javase/downloads/jce8-download-2133166.html>
2. Copy `local_policy.jar` and `US_export_policy.jar` files to the location `JAVA_HOME/jre/lib/security/`. If the files already exist in the destination folder, overwrite them.

## EditFailedException When Releasing Configuration From WebLogic Console

After you upgrade Oracle Identity Manager to 12c (12.2.1.3.0), when you click **Release Configuration** on Oracle WebLogic Console, the following error is seen:

```
weblogic.management.provider.EditFailedException: Error loading jdbc/  
OIMMDS-jdbc.xml
```

This error does not have any functional impact on the WebLogic configuration. To resolve this, open the following DataSource configurations, make any changes, save, and activate the changes:

- ApplicationDB
- mds-oim
- OIMJMSStoreDS

- OIMOperationsDB
- soaOIMLookupDB

## OIM Application Deployment Fails Intermittently

After you upgrade Oracle Identity Manager to 12c (12.2.1.3.0), the oim application deployment may fail with the following error:

```
<Error> <Deployer> <BEA-149231> <Unable to set the activation state to true for the application "oim".
weblogic.application.ModuleException: java.lang.NoClassDefFoundError:
Could
not initialize class oracle.iam.platform.utils.cache.Cache
```

To resolve this, restart the Oracle Identity Manager Server.

## soa-infra Application is in 'Prepared' State Post Upgrade

After you upgrade Oracle Identity Manager (OIM) and Oracle Access Manager (OAM) integrated environment that was set up using Life Cycle Management (LCM) tool, the soa-infra application continues to be in Prepared mode, instead of showing active mode.

To resolve this issue, do the following:

1. Stop all of the managed servers in the private domain. See [Stopping Servers and Processes](#).
2. Take a back up and delete the contents of the private domain.
3. Pack the shared domain and unpack it on the private domain.
4. Start the managed servers in private domain. See [Starting the Servers](#).

Verify that the soa-infra application comes up in active state.

## 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_Iptlif1OpGGi00000U] 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.oimdataprovers.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_Iptlif1OpGGi00000U]
oracle.iam.oimdataprovers.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_Iptlif1OpGGi00000V] 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
```

## SOA Fails to Join Coherence Cluster During the First Start After Upgrade

After you upgrade Oracle Identity Manager (OIM) and Oracle Access Manager (OAM) integrated environment, when you start the Oracle SOA Suite Server for the first time, the coherence cluster fails to start with the following error:

```

<Error> <com.oracle.coherence> <BEA-000000> <2017-08-03
15:49:14.010/123.585 Oracle Coherence GE 12.2.1.3.0
<Error> (thread=Cluster, member=n/a): This member could not join the
cluster because of a mismatch
between Coherence license types. This member was attempting to run in
dev mode.
Rejected by Member(Id=1, Timestamp=2017-08-03 15:36:20.874,

```

```

Address=10.241.57.43:57023, MachineId=8125,
Location=process:19490,member:oam_policy_mgr1, Role=WeblogicServer).>
<Aug 3, 2017 3:49:14,017 PM UTC> <Error> <com.oracle.coherence>
<BEA-000000>
<2017-08-03 15:49:14.017/123.592 Oracle Coherence GE 12.2.1.3.0
<Error> (thread=[ACTIVE] ExecuteThread:
'10' for queue: 'weblogic.kernel.Default (self-tuning)', member=n/a):
Error while starting cluster:
java.lang.RuntimeException: Failed to start Service "Cluster"
(ServiceState=SERVICE_STOPPED, STATE_JOINING)
at
com.tangosol.coherence.component.util.daemon.queueProcessor.Service.star
t(Service.CDB:38)

```

This occurs if both the OIM and OAM WebLogic domains have the same default coherence cluster port. To resolve this issue, change the cluster port for either OAM or OIM by doing the following, pre-upgrade:

1. Log in to the WebLogic Administration console using following URL:  
[http://weblogic\\_admin\\_host:weblogic\\_admin\\_port/console](http://weblogic_admin_host:weblogic_admin_port/console)
2. Click **Environments** on the left navigation pane.
3. Click **Coherence Clusters**, and then click **defaultCoherenceCluster**.
4. Change the port from 7574 to 7575 for either OIM or OAM .

## LDAP User Create and Update Reconciliation Job Fails

LDAP User Create and Update Reconciliation job fails to run with the following exception:

```

java.lang.Exception: Full resync required. Reason: The provided cookie
is older than the start of historical
in the server for the replicated domain : dc=example,dc=com

```

To resolve this issue, you must update the parameter `Last Change Number` of the job. to do this, complete the following steps:

1. Get the value from Oracle Unified Directory using the following command:  

```
./ldapsearch -h <OUHOST>-p 1389 -D "cn=oudadmin" -w Fusionapps1
--control "1.3.6.1.4.1.26027.1.5.4:false:;" -b "cn=changelog"
"(objectclass=*)" "*" +
```
2. Search for the following line in the output of the above command:  

```
changeLogCookie: dc=example,dc=com:0000015dcefd65a3000100000102;
```
3. Fill in `dc=example,dc=com:0000015dcefd65a3000100000102;` in to the `Last Change Number` parameter of the job.



## BI Managed Server is Seen on WebLogic Console After Upgrade

If your 11g Release 2 (11.1.2.3.0) domain had a custom name for BI Managed Server, that is, the name other than `bi_server1`, then this Managed Server will not be deleted during the upgrade.

Post upgrade, the BI managed server lying in Oracle Identity Governance domain is of no use. You can delete this BI server manually by doing the following:

1. Log in to the WebLogic Administration Console using the following URL:

```
http://weblogic_admin_host:weblogic_admin_port/console
```

2. Click **Environments** on the left navigation pane.
3. Click **Servers**.
4. Select the check box against BI Managed Server, and click **Delete**.

## Empty Pages or Panels After Upgrade

After you complete the upgrade, the Applications tile in the OIM Self Service console and the Import/Export links in the Admin console may be rendered as empty pages or panels.

This can occur if the following URIs are being filtered by a proxy:

- `/OIGUI/`
- `/FacadeWebApp/`
- `/iam/`

To avoid this issue, update your proxy rules to allow access to these URIs.

# A

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

Consider that you have a JDK version `jdk1.8.0_121` installed on your machine. When you install and configure an Oracle Fusion Middleware product, the utilities, such as Configuration Wizard (`config.sh|exe`), OPatch, or RCU point to a default JDK, for example, `jdk1.8.0_121`. After some time, Oracle releases a new version of the JDK, say `jdk1.8.0_131` that carries security enhancements and bug fixes. From 12c (12.2.1.3.0) onwards, you can upgrade the existing JDK to a newer version, and can have the complete product stack point to the newer version of the JDK.

You can maintain multiple versions of JDK and switch to the required version on need basis.

- [About Updating the JDK Location After Installing an Oracle Fusion Middleware Product](#)

The binaries and other metadata and utility scripts in the Oracle home and Domain home, such as RCU or Configuration Wizard, use a JDK version that was used while installing the software and continue to refer to the same version of the JDK. The JDK path is stored in a variable called `JAVA_HOME` which is centrally located in `.globalEnv.properties` file inside the `ORACLE_HOME/oui` directory.

## About Updating the JDK Location After Installing an Oracle Fusion Middleware Product

The binaries and other metadata and utility scripts in the Oracle home and Domain home, such as RCU or Configuration Wizard, use a JDK version that was used while installing the software and continue to refer to the same version of the JDK. The JDK path is stored in a variable called `JAVA_HOME` which is centrally located in `.globalEnv.properties` file inside the `ORACLE_HOME/oui` directory.

The utility scripts such as `config.sh|cmd`, `launch.sh`, or `opatch` reside in the `ORACLE_HOME`, and when you invoke them, they refer to the `JAVA_HOME` variable located in `.globalEnv.properties` file. To point these scripts and utilities to the newer version of JDK, you must update the value of the `JAVA_HOME` variable in the `.globalEnv.properties` file by following the directions listed in [Updating the JDK Location in an Existing Oracle Home](#).

To make the scripts and files in your Domain home directory point to the newer version of the JDK, you can follow one of the following approaches:

- Specify the path to the newer JDK on the Domain Mode and JDK screen while running the Configuration Wizard.

For example, consider that you installed Oracle Fusion Middleware Infrastructure with the JDK version 8u191. So while configuring the WebLogic domain with the Configuration Assistant, you can select the path to the newer JDK on the Domain

Mode and JDK screen of the Configuration Wizard. Example: `/scratch/jdk/jdk1.8.0_131`.

- Manually locate the files that have references to the JDK using `grep` (UNIX) or `findstr` (Windows) commands and update each reference. See [Updating the JDK Location in an Existing Oracle Home](#).



#### Note:

If you install the newer version of the JDK in the same location as the existing JDK by overwriting the files, then you don't need to take any action.

When you upgrade Oracle Identity Manager in an integrated environment, you may encounter the OPSS processing error. The following exception is seen when you run `reconfig.sh` command to reconfigure the Oracle Identity Manager domain:

```
SEVERE [93] com.oracle.cie.domain.progress.AbstractProgressGenerator -
Error occurred in
phase {OPSS Processing} execution.
java.lang.IllegalStateException: SecurityContext: Domain Name:
IAMGovernanceDomain
JDBC URL: opss-audit-DBDS:jdbc:oracle:thin:@//slc03rmj:1521/IDMDB
JDBC URL: opss-data-source:jdbc:oracle:thin:@//slc03rmj:1521/idmdb
le.com
Caused by: java.security.InvalidKeyException: Illegal key size
at javax.crypto.Cipher.checkCryptoPerm(Cipher.java:1039)
at javax.crypto.Cipher.implInit(Cipher.java:805)
at javax.crypto.Cipher.chooseProvider(Cipher.java:864)
at javax.crypto.Cipher.init(Cipher.java:1396)
at javax.crypto.Cipher.init(Cipher.java:1327)
```

To resolve this issue:

1. Install the Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files from the following location: [Java Cryptography Extension \(JCE\) Unlimited Strength Jurisdiction Policy Files 8 Download](#).
  2. Copy the `local_policy.jar` and the `US_export_policy.jar` files to the location `JAVA_HOME/jre/lib/security/`. If the files already exist in the destination folder, overwrite them.
- [Updating the JDK Location in an Existing Oracle Home](#)  
The `getProperty.sh|cmd` script displays the value of a variable, such as `JAVA_HOME`, from the `.globalEnv.properties` file. The `setProperty.sh|cmd` script is used to set the value of variables, such as `OLD_JAVA_HOME` or `JAVA_HOME` that contain the locations of old and new JDKs in the `.globalEnv.properties` file.
  - [Updating the JDK Location in an Existing Domain Home](#)  
You must search the references to the current JDK, for example `jdk1.8.0_121` manually, and replace those instances with the location of the new JDK.

## Updating the JDK Location in an Existing Oracle Home

The `getProperty.sh|cmd` script displays the value of a variable, such as `JAVA_HOME`, from the `.globalEnv.properties` file. The `setProperty.sh|cmd` script is used to set the value of variables, such as `OLD_JAVA_HOME` or `JAVA_HOME` that contain the locations of old and new JDKs in the `.globalEnv.properties` file.

The `getProperty.sh|cmd` and `setProperty.sh|cmd` scripts are located in the following location:

(UNIX) `ORACLE_HOME/oui/bin`

(Windows) `ORACLE_HOME\oui\bin`

Where, `ORACLE_HOME` is the directory that contains the products using the current version of the JDK, such as `jdk1.8.0_121`.

To update the JDK location in the `.globalEnv.properties` file:

1. Use the `getProperty.sh|cmd` script to display the path of the current JDK from the `JAVA_HOME` variable. For example:

(UNIX) `ORACLE_HOME/oui/bin/getProperty.sh JAVA_HOME`

(Windows) `ORACLE_HOME\oui\bin\getProperty.cmd JAVA_HOME`

`echo JAVA_HOME`

Where `JAVA_HOME` is the variable in the `.globalEnv.properties` file that contains the location of the JDK.

2. Back up the path of the current JDK to another variable such as `OLD_JAVA_HOME` in the `.globalEnv.properties` file by entering the following commands:

(UNIX) `ORACLE_HOME/oui/bin/setProperty.sh -name OLD_JAVA_HOME -value specify_the_path_of_current_JDK`

(Windows) `ORACLE_HOME\oui\bin\setProperty.cmd -name OLD_JAVA_HOME -value specify_the_path_of_current_JDK`

This command creates a new variable called `OLD_JAVA_HOME` in the `.globalEnv.properties` file, with a value that you have specified.

3. Set the new location of the JDK in the `JAVA_HOME` variable of the `.globalEnv.properties` file, by entering the following commands:

(UNIX) `ORACLE_HOME/oui/bin/setProperty.sh -name JAVA_HOME -value specify_the_location_of_new_JDK`

(Windows) `ORACLE_HOME\oui\bin\setProperty.cmd -name JAVA_HOME -value specify_the_location_of_new_JDK`

After you run this command, the `JAVA_HOME` variable in the `.globalEnv.properties` file now contains the path to the new JDK, such as `jdk1.8.0_131`.

## Updating the JDK Location in an Existing Domain Home

You must search the references to the current JDK, for example `jdk1.8.0_121` manually, and replace those instances with the location of the new JDK.

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

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

(UNIX) *DOMAIN\_HOME*/bin/setNMJavaHome.sh

(Windows) *DOMAIN\_HOME*\bin\setNMJavaHome.cmd

(UNIX) *DOMAIN\_HOME*/nodemanager/nodemanager.properties

(Windows) *DOMAIN\_HOME*\nodemanager\nodemanager.properties

(UNIX) *DOMAIN\_HOME*/bin/setDomainEnv.sh

(Windows) *DOMAIN\_HOME*\bin\setDomainEnv.cmd