

Oracle® Fusion Middleware

Upgrading to the Oracle Fusion Middleware Infrastructure



12c (12.2.1.3.0)

E95622-03

November 2020

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

ORACLE®

Oracle Fusion Middleware Upgrading to the Oracle Fusion Middleware Infrastructure, 12c (12.2.1.3.0)

E95622-03

Copyright © 2014, 2020, Oracle and/or its affiliates.

Primary Author: Oracle Corporation

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

What's New in This Guide

New and Changed Features for 12c (12.2.1.3.0)	x
---	---

1 Introduction to Upgrading Oracle Fusion Middleware Infrastructure to 12c (12.2.1.3.0)

About the Starting Points for an Oracle Fusion Middleware Infrastructure Upgrade	1-2
About Oracle Fusion Middleware Infrastructure 12c	1-3
Key Differences Between Application Developer 11g and Infrastructure 12c	1-3
Infrastructure 12c Includes Oracle WebLogic Server	1-3
Database Schema Requirement for Infrastructure 12c	1-4
Using an OID-based Policy Store	1-5
Infrastructure 12c Domains Can Include Oracle HTTP Server	1-5
Upgrading Security Store	1-6
About Oracle Fusion Middleware Infrastructure Standard Topology	1-6
Oracle Fusion Middleware Infrastructure Standard Upgrade Topology	1-7
About the New Features for Oracle Fusion Middleware Infrastructure 12c	1-10
About Upgrade Restrictions	1-10
Upgrading Custom Applications Using Oracle JDeveloper 12c	1-11

2 Pre-Upgrade Requirements

Cloning Your Production Environment for Testing	2-2
Maintaining Your Custom setDomainEnv Settings	2-3
Using No-Auth SSL Mode in OID Security Store	2-4
Removing the Server Instance Scope from OWSM Policy Sets	2-4

Cloning Predefined Documents and Migrating OWSM Policy Attachments	2-5
Pre-Upgrade Checklist	2-5
Creating a Complete Backup	2-7
Backing Up the Schema Version Registry Table	2-7
Maintaining Customized Domain and Environment Settings	2-8
Verifying Certification and System Requirements	2-9
Verify Your Environment Meets Certification Requirements	2-9
Verify System Requirements and Specifications	2-10
Migrating from a 32-Bit to a 64-Bit Operating System	2-10
Verify That the Database Hosting Oracle Fusion Middleware is Supported	2-14
Verify That the JDK Is Certified for This Release of Oracle Fusion Middleware	2-14
Updating Policy Files when Using Enhanced Encryption (AES 256)	2-15
Purging Unused Data	2-15
Creating an Edition on the Server for Edition-Based Redefinition	2-15
Creating a Non-SYSDBA User to Run the Upgrade Assistant	2-16

3 Upgrading Oracle Fusion Middleware Infrastructure from 11g Release

About the Oracle Fusion Middleware Infrastructure Upgrade Process (11g to 12c)	3-3
Installing Oracle Fusion Middleware Infrastructure	3-5
Running a Pre-Upgrade Readiness Check	3-7
About Running a Pre-Upgrade Readiness Check	3-7
Starting the Upgrade Assistant in Readiness Mode	3-8
Upgrade Assistant Parameters	3-9
Performing a Readiness Check with the Upgrade Assistant	3-10
Understanding the Readiness Report	3-12
Stopping Servers and Processes	3-16
Upgrading Product Schemas	3-17
Starting the Upgrade Assistant	3-18
Upgrade Assistant Parameters	3-18
Creating and Upgrading Schemas Using the Upgrade Assistant	3-20
Verifying the Schema Upgrade	3-23
Creating the Required 12c Schemas with the RCU	3-24
About Reconfiguring the Domain	3-28
Backing Up the Domain	3-30
Starting the Reconfiguration Wizard	3-31
Reconfiguring the Domain with the Reconfiguration Wizard	3-32
Upgrading Domain Component Configurations	3-37
Starting the Upgrade Assistant	3-38
Upgrade Assistant Parameters	3-38

Upgrading the Domain Configurations with the Upgrade Assistant	3-40
Verifying the Domain-Specific-Component Configurations Upgrade	3-42
Starting Servers and Processes	3-43
Using the Upgrade Validation Checklist	3-44
Reapplying Custom Configuration Settings to setDomainEnv	3-45
Configuring an Oracle Fusion Middleware 12c Audit Data Store	3-45
Maintaining the Security Status of Older Java EE Web Service Applications	3-46
Documentation Resources for Managing your Oracle Fusion Middleware 12c Software	3-46
Using Your 11g Application Deployments in Oracle Fusion Middleware 12c	3-47
About Oracle Application Development Framework (ADF) 12c	3-47
About Oracle JDeveloper 12c	3-48
Installing Oracle JDeveloper 12c	3-48
Migrating Applications Using Oracle JDeveloper 12c	3-48
About Migrating Asynchronous Web Services with Oracle JDeveloper 12c	3-48
If Your Existing Environment is a Clustered Configuration...	3-49
Packing the Domain on the Primary Node	3-49
Unpacking the Domain on the Secondary Node	3-49
Starting Servers and Processes	3-49

4 Upgrading Oracle Fusion Middleware Infrastructure from a Previous 12c Release

About the Oracle Fusion Middleware Infrastructure Upgrade Process (from a Previous 12c Release)	4-2
Installing Oracle Fusion Middleware Infrastructure	4-5
Running a Pre-Upgrade Readiness Check	4-7
About Running a Pre-Upgrade Readiness Check	4-7
Starting the Upgrade Assistant in Readiness Mode	4-8
Upgrade Assistant Parameters	4-8
Performing a Readiness Check with the Upgrade Assistant	4-10
Understanding the Readiness Report	4-12
Stopping Servers and Processes	4-16
Upgrading Product Schemas	4-16
Starting the Upgrade Assistant	4-17
Upgrade Assistant Parameters	4-17
Upgrading Schemas Using the Upgrade Assistant	4-19
Verifying the Schema Upgrade	4-22
About Reconfiguring the Domain	4-23
Backing Up the Domain	4-25
Starting the Reconfiguration Wizard	4-26
Reconfiguring the Domain with the Reconfiguration Wizard	4-27

Upgrading Domain Component Configurations	4-32
Starting the Upgrade Assistant	4-33
Upgrade Assistant Parameters	4-33
Upgrading the Domain Configurations with the Upgrade Assistant	4-35
Verifying the Domain-Specific-Component Configurations Upgrade	4-37
Starting Servers and Processes	4-38
Using the Upgrade Validation Checklist	4-39
Reapplying Custom Configuration Settings to setDomainEnv	4-40
Maintaining the Security Status of Older Java EE Web Service Applications	4-40
Documentation Resources for Managing your Oracle Fusion Middleware 12c Software	4-41
If Your Existing Environment is a Clustered Configuration...	4-42
Packing the Domain on the Primary Node	4-42
Unpacking the Domain on the Secondary Node	4-42
Starting Servers and Processes	4-42

5 Tasks to Perform After Upgrade

Using the Upgrade Validation Checklist	5-2
Verifying the Domain-Specific-Component Configurations Upgrade	5-2
Reapplying Custom Configuration Settings to setDomainEnv	5-3
Configuring an Oracle Fusion Middleware 12c Audit Data Store	5-3
Maintaining the Security Status of Older Java EE Web Service Applications	5-3
Documentation Resources for Managing your Oracle Fusion Middleware 12c Software	5-4
Using Your 11g Application Deployments in Oracle Fusion Middleware 12c	5-5
About Oracle Application Development Framework (ADF) 12c	5-5
About Oracle JDeveloper 12c	5-6
Installing Oracle JDeveloper 12c	5-6
Migrating Applications Using Oracle JDeveloper 12c	5-6
About Migrating Asynchronous Web Services with Oracle JDeveloper 12c	5-6

A Troubleshooting the Infrastructure Upgrade

Authentication Failure — JSchException: Auth Fail	A-1
Error while Copying User Messaging Service (UMS) Configuration Files	A-2

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

About Updating the JDK Location After Installing an Oracle Fusion Middleware Product	B-1
--	-----

Updating the JDK Location in an Existing Oracle Home	B-2
Updating the JDK Location in an Existing Domain Home	B-3

Preface

This document describes how to upgrade an existing Oracle Fusion Middleware Infrastructure environment to 12c (12.2.1.3.0).

- [Audience](#)
- [Documentation Accessibility](#)
- [Related Documents](#)

Upgrade documentation is organized by tasks in the 12c documentation library. The task-specific pages provide direct links to common upgrade procedures and related documentation.
- [Conventions](#)

Audience

This document is intended for administrators who are familiar with Oracle Fusion Middleware installation, upgrade, and administration tasks.

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

Upgrade documentation is organized by tasks in the 12c documentation library. The task-specific pages provide direct links to common upgrade procedures and related documentation.

You can refer the Oracle Fusion Middleware Library for additional information.

- For installation information, see Fusion Middleware Installation Documentation.
- For upgrade information, see Fusion Middleware 12c Upgrade Documentation.
- For administration-related information, see Fusion Middleware 12c Administration Documentation.

- For release-related information, see Fusion Middleware 12c Release Notes.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<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.

What's New in This Guide

This topic introduces the new and changed features of the Oracle Fusion Middleware upgrade tools and processes.

It also provides information about this book and provides pointers to additional information.

- [New and Changed Features for 12c \(12.2.1.3.0\)](#)

New and Changed Features for 12c (12.2.1.3.0)

Oracle Fusion Middleware 12c introduces the following new upgrade processes and tools, as well as a new set of installers, which are now referred to as distributions:

- As of release 12c (12.1.2), there is a new distribution called Oracle Fusion Middleware Infrastructure. This distribution includes an installer that combines the features and capabilities of the Oracle WebLogic Server and Application Developer 11g installers.

As a result, this distribution contains everything required to create Oracle WebLogic Server domains that can be used to deploy and manage Java and Oracle Application Development Framework (Oracle ADF) applications.

See *Overview of Oracle Fusion Middleware* in *Understanding Oracle Fusion Middleware*.
- As of release 12c (12.1.2), there is a new and improved Oracle Fusion Middleware Upgrade Assistant, which is used to upgrade the Oracle Fusion Middleware database schemas and upgrade the component configurations to 12c.

See *Understanding the Upgrade and Configuration Tools* in *Planning an Upgrade of Oracle Fusion Middleware*.
- As of release 12c (12.1.2), there is a new Reconfiguration Wizard, which upgrades your existing 11g domains to 12c.

See *Reconfiguring WebLogic Domains* in *Upgrading Oracle WebLogic Server*.
- As of release 12c (12.2.1.3.0), you can create and upgrade the required schemas with the Upgrade Assistant.

1

Introduction to Upgrading Oracle Fusion Middleware Infrastructure to 12c (12.2.1.3.0)

Upgrading to Oracle Fusion Middleware Infrastructure 12c requires careful preparation and planning. Oracle provides tools and technology to automate much of the upgrade process.

Following the pre-upgrade information and preparation steps, the procedures in this guide are divided into two sections:

- Upgrade from 11g release is described in [Upgrading Oracle Fusion Middleware Infrastructure from 11g Release](#)
- Upgrade from 12c release is described in [Upgrading Oracle Fusion Middleware Infrastructure from a Previous 12c Release](#)

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 12c Upgrade Tasks](#)

The following topics describe the concepts related to upgrading Oracle Fusion Middleware Infrastructure:

- [About the Starting Points for an Oracle Fusion Middleware Infrastructure Upgrade](#)
The primary focus of the upgrade procedures in this guide is to upgrade an existing Application Server 11g domain or the existing 12c (12.1.2.0.0, 12.1.3.0.0, 12.2.1.0.0, 12.2.1.1.0, and 12.2.1.2.0) domains, and the Oracle Fusion Middleware component configurations in that domain to Oracle Fusion Middleware Infrastructure 12c (12.2.1.3.0).
- [Upgrading Security Store](#)
Before upgrading the OPSS security store, it is important to back up the security store so that it can be recovered in case the upgrade fails.
- [About Oracle Fusion Middleware Infrastructure Standard Topology](#)
The steps to upgrade Oracle Fusion Middleware Infrastructure to 12c (12.2.1.3.0) depend on the existing production topology (either 11g or a previous 12c release).

- [About the New Features for Oracle Fusion Middleware Infrastructure 12c](#)
Several changes have been made to Oracle Fusion Middleware Infrastructure 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.
- [Upgrading Custom Applications Using Oracle JDeveloper 12c](#)
If you have deployed custom applications to an Oracle Fusion Middleware Application Developer 11g domain, then the application deployments should function as they did in Oracle Fusion Middleware 11g after the upgrade procedure is complete. However, if you want to take advantage of new Oracle Application Development Framework (Oracle ADF) 12c features, download and install Oracle JDeveloper 12c.

About the Starting Points for an Oracle Fusion Middleware Infrastructure Upgrade

The primary focus of the upgrade procedures in this guide is to upgrade an existing Application Server 11g domain or the existing 12c (12.1.2.0.0, 12.1.3.0.0, 12.2.1.0.0, 12.2.1.1.0, and 12.2.1.2.0) domains, and the Oracle Fusion Middleware component configurations in that domain to Oracle Fusion Middleware Infrastructure 12c (12.2.1.3.0).

You can upgrade to Oracle Fusion Middleware Infrastructure 12c (12.2.1.3.0) from the supported 11g and 12c releases:

- Oracle Fusion Middleware Application Developer 11g (11.1.1.7, 11.1.1.8, and 11.1.1.9)
- Oracle Fusion Middleware Infrastructure 12c (12.1.2.0.0, 12.1.3.0.0, 12.2.1.0.0, 12.2.1.1.0, and 12.2.1.2.0)

Note:

If you are upgrading Oracle HTTP Server instances associated with an existing Application Developer 11g domain or instructions for upgrading a standalone Oracle HTTP Server, see *Upgrading a Standalone Oracle HTTP Server from 11g to 12c* in *Upgrading Oracle HTTP Server*.

- [About Oracle Fusion Middleware Infrastructure 12c](#)
Oracle Fusion Middleware Infrastructure distribution, which is available as part of the Oracle Fusion Middleware 12c (12.2.1.3.0) release, provides a set of technologies and components similar to those provided by the Oracle WebLogic Server and Application Developer installers in 11g. The Infrastructure distribution combines the WebLogic Server and the Java Required Files (JRF) that are required to install other Fusion Middleware products.

- [Key Differences Between Application Developer 11g and Infrastructure 12c](#)
Oracle Fusion Middleware Infrastructure 12c is similar to the 11g Application Developer installation, except for few differences listed in this section.

About Oracle Fusion Middleware Infrastructure 12c

Oracle Fusion Middleware Infrastructure distribution, which is available as part of the Oracle Fusion Middleware 12c (12.2.1.3.0) release, provides a set of technologies and components similar to those provided by the Oracle WebLogic Server and Application Developer installers in 11g. The Infrastructure distribution combines the WebLogic Server and the Java Required Files (JRF) that are required to install other Fusion Middleware products.

Key Differences Between Application Developer 11g and Infrastructure 12c

Oracle Fusion Middleware Infrastructure 12c is similar to the 11g Application Developer installation, except for few differences listed in this section.

- [Infrastructure 12c Includes Oracle WebLogic Server](#)
The Application Developer 11g installation required two separate installations (Oracle WebLogic Server and then Application Developer to add the Oracle JRF libraries and components). In Oracle Fusion Middleware 12c, a fresh installation requires only the Oracle Fusion Middleware Infrastructure distribution, which contains both Oracle WebLogic Server and the Java Required Files (JRF).
- [Database Schema Requirement for Infrastructure 12c](#)
Unlike the Application Developer 11g installation, the Infrastructure 12c installation requires you to create a set of required schemas in a supported database. In particular, you must use the 12c Repository Creation Utility (RCU) to create the required database schemas before you can configure the Oracle Fusion Middleware Infrastructure 12 c software.
- [Using an OID-based Policy Store](#)
If you are using an Oracle Internet Directory (OID)-based policy store in 11g, use the 12c Repository Creation Utility (RCU) to create the new 12c OPSS schema. In the Upgrade Assistant, select the OPSS schema; the Upgrade Assistant upgrades the OID-based policy store. You do not need to reassociate an OID-based policy store to upgrade it to 12c.
- [Infrastructure 12c Domains Can Include Oracle HTTP Server](#)
In Oracle Fusion Middleware 11g, Oracle HTTP Server instances are typically configured in a separate Oracle instance directory outside the 11g Middleware home. However, in Oracle Fusion Middleware 12c, Oracle HTTP Server instances can be configured a part of an Oracle WebLogic Server domain, using the Oracle Fusion Middleware Configuration Wizard.

Infrastructure 12c Includes Oracle WebLogic Server

The Application Developer 11g installation required two separate installations (Oracle WebLogic Server and then Application Developer to add the Oracle JRF libraries and components). In Oracle Fusion Middleware 12c, a fresh installation requires only the Oracle Fusion Middleware Infrastructure distribution, which contains both Oracle WebLogic Server and the Java Required Files (JRF).

Note that the upgrade procedure does not require the configuration of a new Oracle Fusion Middleware 12c domain. Instead, you use the Reconfiguration Wizard to upgrade the domain. See Understanding Upgrade and Configuration Tools in *Planning an Upgrade of Oracle Fusion Middleware*.

Database Schema Requirement for Infrastructure 12c

Unlike the Application Developer 11g installation, the Infrastructure 12c installation requires you to create a set of required schemas in a supported database. In particular, you must use the 12c Repository Creation Utility (RCU) to create the required database schemas before you can configure the Oracle Fusion Middleware Infrastructure 12 c software.

Depending upon your requirements, you must install one or more of the following database schemas before you can upgrade to Oracle Fusion Middleware Infrastructure 12c:

- The Service Table (STB) schema is a new schema required for all Oracle Fusion Middleware Infrastructure 12c installations. This schema enables a new 12c feature called Cross-Component Wiring. See Wiring Components to Work Together in *Administering Oracle Fusion Middleware*. If you are upgrading from a 11g release, then you must create the 12c STB schemas with the Repository Creation Utility (RCU) . However, if you are upgrading from a previous 12c release, you need not create the STB schemas; you can upgrade them with the Upgrade Assistant.
- The OPSS schema, which provides a database-based policy store for Oracle Platform Security Services.

[Table 1-1](#) provides an overview of the policy stores used in 11g and how to upgrade them.

Table 1-1 Overview of Upgrading the Oracle Fusion Middleware 11g Policy Store

If you are using...	Then perform the following action:
Oracle Internet Directory (OID)-based policy store in 11g	<ul style="list-style-type: none"> – Before the upgrade, use the 12c Repository Creation Utility (RCU) to create the new 12c OPSS schema. – In the Upgrade Assistant, select the OPSS schema; the Upgrade Assistant upgrades the OID-based policy store.
File-based policy store and the OPSS schema in 11g	Use the RCU to create the 12c OPSS schema.
Database-based policy store and the OPSS schema	Use the Upgrade Assistant to upgrade the existing 11g OPSS schema.

- The IAU schema, which is used for the OPSS auditing capabilities.
You might be required to create the IAU 12cschema, depending on whether or not you are using an Audit Data Store in 11g and the type of Audit Data Store you are using.

Table 1-2 Overview of Upgrading the Oracle Fusion Middleware 11g Audit Store

If you are using...	Then perform the following action:
File-based audit store in 11g	– Before the upgrade, create a new 12c IAU schema.

Table 1-2 (Cont.) Overview of Upgrading the Oracle Fusion Middleware 11g Audit Store

If you are using...	Then perform the following action:
Database-based audit store and the IAU 11g schema	– Use the Upgrade Assistant to upgrade the existing 11g IAU schema, and then reference the upgraded 11g schema during the domain reconfiguration.

Using an OID-based Policy Store

If you are using an Oracle Internet Directory (OID)-based policy store in 11g, use the 12c Repository Creation Utility (RCU) to create the new 12c OPSS schema. In the Upgrade Assistant, select the OPSS schema; the Upgrade Assistant upgrades the OID-based policy store. You do not need to reassociate an OID-based policy store to upgrade it to 12c.

 **Note:**

The 12c OPSS database schema is required only so you can reference the 12c schema during the reconfiguration of the domain. Your domain will continue to use the OID-based policy store after the upgrade.

Infrastructure 12c Domains Can Include Oracle HTTP Server

In Oracle Fusion Middleware 11g, Oracle HTTP Server instances are typically configured in a separate Oracle instance directory outside the 11g Middleware home. However, in Oracle Fusion Middleware 12c, Oracle HTTP Server instances can be configured a part of an Oracle WebLogic Server domain, using the Oracle Fusion Middleware Configuration Wizard.

Oracle HTTP Server 11g instances are managed using the Oracle Process Manager and Notification Server (OPMN) management software. Optionally, the Oracle HTTP Server 11g instances can be "associated with" the WebLogic domain.

When configured as part of an Oracle Fusion Middleware Infrastructure domain, Oracle HTTP Server instances can be managed using Oracle Enterprise Manager Fusion Middleware Control and the Oracle WebLogic Scripting Tool (WLST). In Oracle Fusion Middleware 12c, the Node Manager agent is responsible for delegating and executing management requests to OHS instances.

For more information about the changes to the ways system components, such as Oracle HTTP Server, are configured and managed in Oracle Fusion Middleware 12c, as well as other key changes for Oracle Fusion Middleware 12c, see [About the New Features for Oracle Fusion Middleware Infrastructure 12c](#).

To upgrade the Oracle HTTP Server, see Introduction to Upgrading Oracle HTTP Server to 12c (12.2.1.3.0) in *Upgrading Oracle HTTP Server*.

Upgrading Security Store

Before upgrading the OPSS security store, it is important to back up the security store so that it can be recovered in case the upgrade fails.

For details about backing up the security store, see [Backing Up and Recovering the OPSS Security Store](#).

The upgrade procedure varies depending on the type of security store you start from. The security store to be upgraded can be file-based, OID-based, or DB-based. Note that the procedures vary depending upon the type of source audit data store (file-based or DB-based).

For more information about upgrading security stores, see [Upgrading Security to 12.2.1.3.0](#).

About Oracle Fusion Middleware Infrastructure Standard Topology

The steps to upgrade Oracle Fusion Middleware Infrastructure to 12c (12.2.1.3.0) depend on the existing production topology (either 11g or a previous 12c release).

Using the Oracle Fusion Middleware Application Developer 11g software, you can create a variety of production topologies to suit the needs of your applications, your organization, and your application users.

It is difficult to provide exact upgrade instructions for every possible Application Developer 11g installation. To solve this problem, this upgrade documentation provides detailed instructions for upgrading a typical Application Developer configuration. This typical topology is referred to as the *standard upgrade topology*.

Specifically, for the purposes of this guide, it is assumed that you have used Oracle WebLogic Server and the Application Developer 11g software to configure a domain that contains a cluster of two managed servers, both of which are configured to support Oracle JRF and the deployment of Oracle ADF applications.

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

Note:

For additional information about the upgrade process and planning resources to ensure your upgrade is successful, see [Preparing to Upgrade in *Planning an Upgrade of Oracle Fusion Middleware*](#).

If you are upgrading from a previous 12c release, the standard topology remains unchanged. If you are upgrading from 11g, refer to the following section to upgrade Oracle Fusion Middleware Infrastructure standard topology:

- [Oracle Fusion Middleware Infrastructure Standard Upgrade Topology](#)
When upgrading from 11g to 12c (12.2.1.3.0), the Oracle Fusion Middleware Infrastructure topology is updated, as illustrated by the standard upgrade topology

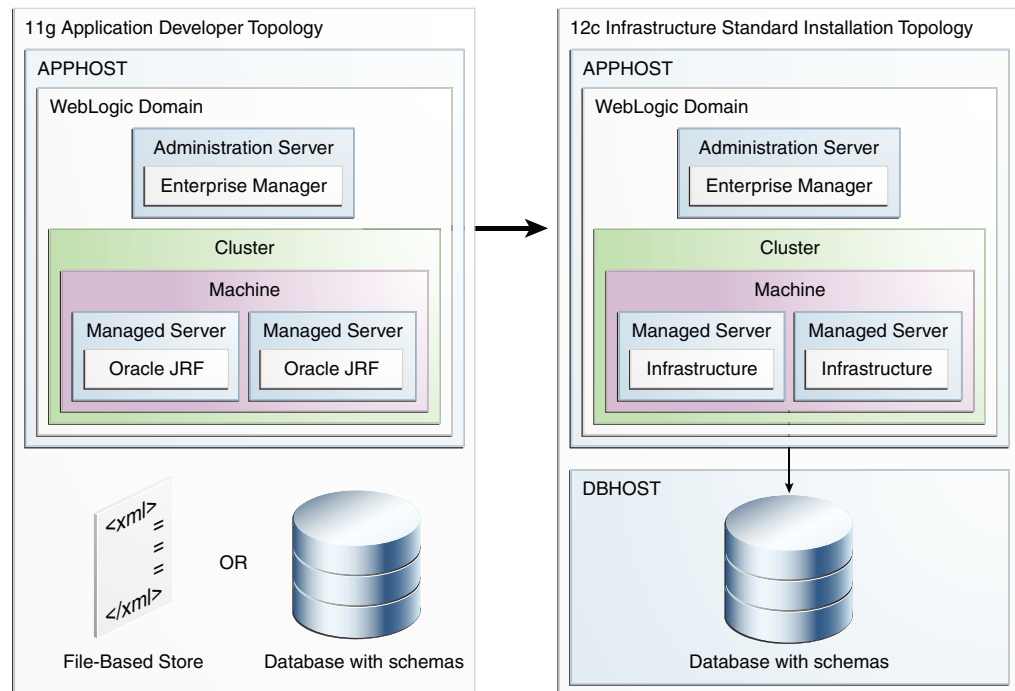
described here. If you are upgrading from a previous 12c release, the standard topology remains unchanged.

Oracle Fusion Middleware Infrastructure Standard Upgrade Topology

When upgrading from 11g to 12c (12.2.1.3.0), the Oracle Fusion Middleware Infrastructure topology is updated, as illustrated by the standard upgrade topology described here. If you are upgrading from a previous 12c release, the standard topology remains unchanged.

[Figure 1-1](#) shows the Oracle Fusion Middleware 11g Application Developer standard upgrade topology and the resulting Oracle Fusion Middleware 12c Infrastructure topology as it appears after you complete the upgrade procedures in this guide.

Figure 1-1 Oracle Fusion Middleware Infrastructure Standard Upgrade Topology



All elements in this topology illustration are described in [Table 1-3](#).

Table 1-3 Description of the Elements in the Oracle Fusion Middleware Infrastructure Standard Upgrade Topology

Element	Description and Links to Additional Documentation
11g Application Developer Topology	<p>This is the label for the left side of Figure 1-1. It shows a typical single-host topology created using the Oracle Fusion Middleware 11g Application Developer installer. It consists of a single domain that contains a cluster of two managed servers and the Administration Server. It also has an optional file-based store or database with schemas.</p> <p>This document describes, step-by-step, how to upgrade this topology to an equivalent topology created using the Oracle Fusion Middleware 12c Infrastructure distribution.</p>
12c Infrastructure Standard Installation Topology	<p>This is the label for the right side of the figure. It shows a typical single-host topology created using the Oracle Fusion Middleware 12c Infrastructure distribution.</p> <p>Like the Application Developer 11g topology, it also consists of a single domain that contains a cluster of two managed servers and the Administration Server.</p>
APPHOST	<p>Standard term used in Oracle documentation referring to the machine that is hosting the application tier.</p>
DBHOST	<p>Standard term used in Oracle documentation referring to the machine that is hosting the database.</p> <p>Note that for Application Developer 11g, a database is optional; for Oracle Fusion Middleware 12c, a database is required. See Database Schema Requirement for Infrastructure 12c.</p>
File-Based Store	<p>An XML, file-based security store. In 11g, you could use a file-base security store or a database-based security store.</p> <p>However, in 12c, the file-based store is deprecated, and you must use the Oracle Platform Security Services (OPSS) schema in a supported database. The Reconfiguration Wizard automates the process of reassociating file-based stores to DB-based stores.</p>
Database with Schemas	<p>Represents a supported database, where the Oracle Fusion Middleware schemas have been created using the Repository Creation Utility (RCU).</p>
WebLogic Domain	<p>A logically related group of Java components (in this case, the administration Server, Managed Servers, and other related software components).</p> <p>See <i>What is an Oracle WebLogic Server Domain</i> in <i>Understanding Oracle Fusion Middleware</i>.</p>
Administration Server	<p>The central control entity of a domain which maintains the domain's configuration objects and distributes configuration changes to Managed Servers.</p> <p>See <i>What is the Administration Server</i> in <i>Understanding Oracle Fusion Middleware</i>.</p>

Table 1-3 (Cont.) Description of the Elements in the Oracle Fusion Middleware Infrastructure Standard Upgrade Topology

Element	Description and Links to Additional Documentation
Enterprise Manager	<p>Oracle Enterprise Manager Fusion Middleware Control. This is the main tool that can be used to manage your domain.</p> <p>See Oracle Enterprise Manager Fusion Middleware Control in <i>Understanding Oracle Fusion Middleware</i>.</p>
Cluster	<p>A collection of multiple WebLogic Server instances running simultaneously and working together.</p> <p>See Understanding Managed Servers and Managed Server Clusters in <i>Understanding Oracle Fusion Middleware</i>.</p>
Machine	<p>Logical representation of the computer that hosts one or more WebLogic Server instances (servers). Machines are also the logical glue between WebLogic Managed Servers and the Node Manager; in order to start or stop a Managed Server with Node Manager, the Managed Server must be associated with a machine.</p>
Managed Server	<p>Host for your applications, application components, Web services, and their associated resources.</p> <p>See Understanding Managed Servers and Managed Server Clusters in <i>Understanding Oracle Fusion Middleware</i>.</p>
Oracle JRF	<p>Oracle JRF (Java Required Files) consists of those components not included in the Oracle WebLogic Server installation and that provide common functionality for Oracle business applications and application frameworks.</p> <p>JRF consists of several independently developed libraries and applications that are deployed into a common location. The components that are considered part of Java Required Files include Oracle Application Development Framework shared libraries and ODL logging handlers.</p>
Infrastructure	<p>Oracle Fusion Middleware 12c term (similar to Oracle JRF) that refers to the collection of services that include the following:</p> <ul style="list-style-type: none"> <li data-bbox="857 1482 1378 1667">• Metadata repository (MDS) This contains metadata for Oracle Fusion Middleware components, such as the Oracle Application Developer Framework. See What is the Metadata Repository in <i>Understanding Oracle Fusion Middleware</i>. <li data-bbox="857 1675 1409 1703">• Oracle Application Developer Framework (ADF) <li data-bbox="857 1711 1333 1738">• Oracle Web Services Manager (OWSM)

The Application Developer 11g topology is similar to the Oracle Fusion Middleware Infrastructure 12c topology, except for the differences described in [Key Differences Between Application Developer 11g and Infrastructure 12c](#).

About the New Features for Oracle Fusion Middleware Infrastructure 12c

Several changes have been made to Oracle Fusion Middleware Infrastructure between 11g and 12c.

In particular, refer to the following sections in *Understanding Oracle Fusion Middleware*:

- New and Changed Features for 12c
- New and Deprecated Terminology for 12c
- What is the WebLogic Management Framework?

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 (12.2.1.3.0) 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 in Understanding Interoperability and Compatibility*.

When performing the upgrade of your hardware or software, verify that your Oracle Fusion Middleware software is certified to support the new operating system or computer hardware. For more information, refer to the following resources:

- Oracle Fusion Middleware Supported System Configurations
- Oracle® Fusion Middleware System Requirements and Specifications

Upgrading Custom Applications Using Oracle JDeveloper 12c

If you have deployed custom applications to an Oracle Fusion Middleware Application Developer 11g domain, then the application deployments should function as they did in Oracle Fusion Middleware 11g after the upgrade procedure is complete. However, if you want to take advantage of new Oracle Application Development Framework (Oracle ADF) 12c features, download and install Oracle JDeveloper 12c.

See Installing the Oracle JDeveloper Software in *Installing Oracle JDeveloper*.

To test your 11g applications in preparation for an upgrade to Oracle Fusion Middleware Infrastructure 12c, open your existing Oracle JDeveloper 11g projects in Oracle JDeveloper 12c. Oracle JDeveloper migrates the projects to 12c. You can then test your applications with the embedded application server that is available from within Oracle JDeveloper. After you have reviewed and optionally modified your applications in Oracle JDeveloper 12c, upgrade the Application Developer 11g domain to Oracle Fusion Middleware Infrastructure 12c and redeploy the applications.

For more information about migrating your applications, see Migrating Oracle JDeveloper From a Previous Version in *Installing Oracle JDeveloper*.

2

Pre-Upgrade Requirements

Before you begin to upgrade Oracle Fusion Middleware Infrastructure 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.

- [Pre-Upgrade Checklist](#)
The Pre-Upgrade Checklist identifies tasks that can be performed before you begin your 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 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 then (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 an Edition on the Server for Edition-Based Redefinition](#)
Before upgrading an Edition-Based Redefinition (EBR) enabled schema, you must connect to the database server and create an edition on the database server for 12c (12.2.1.3.0).
- [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.
- [Maintaining Your Custom `setDomainEnv` Settings](#)
Every domain includes dynamically generated domain and server startup scripts, such as `setDomainEnv`. Oracle recommends that you do not modify these startup scripts, as any changes you make to them will be overwritten during subsequent domain upgrade operations.
- [Using No-Auth SSL Mode in OID Security Store](#)
The SSL protocol provides transport layer security with authenticity, integrity, and confidentiality, for a connection between a client and server. The SSL authentication mode is controlled by the attribute `orclsslauthentication` in

the instance-specific configuration entry. By default, Oracle Internet Directory (OID) uses SSL No Authentication Mode (`orclsslauthentication=1`).

- [Removing the Server Instance Scope from OWSM Policy Sets](#)
The Server Instance Scope in policy sets was not recommended in 11g (11.1.1.7.0) and is not supported in 12c. However, if you have policy sets that use the Server Instance Scope, they are disabled during the upgrade to 12c. Therefore, you must remove the Server Instance Scope from all the 11g policy sets before upgrading to 12c.
- [Cloning Predefined Documents and Migrating OWSM Policy Attachments](#)
When upgrading, it is important to note that any predefined documents that have not been customized for your environment are replaced with read-only versions, and new, predefined, read-only documents are added. However, any existing predefined documents that have been customized and any user-created custom policies in the repository are not overwritten.

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 then (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; 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.

Maintaining Your Custom `setDomainEnv` Settings

Every domain includes dynamically generated domain and server startup scripts, such as `setDomainEnv`. Oracle recommends that you do not modify these startup scripts, as any changes you make to them will be overwritten during subsequent domain upgrade operations.

 **Caution:**

Changes made to the `setDomainEnv` script - or any other startup script - before an upgrade are overwritten by the new, regenerated scripts during the domain reconfiguration process. Create a separate file to store your customized domain settings before you upgrade.

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 java command line options for running the servers, or specify additional environment variables, for instance. Any custom settings you add to this file are preserved during domain upgrade operation and are carried over to the remote servers when using the `pack` and `unpack` commands.

Following is an example of 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.

Using No-Auth SSL Mode in OID Security Store

The SSL protocol provides transport layer security with authenticity, integrity, and confidentiality, for a connection between a client and server. The SSL authentication mode is controlled by the attribute `orclsslauthentication` in the instance-specific configuration entry. By default, Oracle Internet Directory (OID) uses SSL No Authentication Mode (`orclsslauthentication=1`).

If you are upgrading to 12c Infrastructure, and using OID as the security policy store with Oracle WebLogic Server, then you may need to modify the default SSL mode. In Oracle Internet Directory 11g, SSL interoperability mode is enabled by default. But Oracle Internet Directory is fully compliant with the JDK's SSL, provided SSL interoperability mode is disabled.

The default use of No-Auth SSL mode in Oracle Internet Directory (OID) is discouraged for production environments due to the susceptibility to Man-in-the-Middle (MITM) attacks.

However, if No-Auth SSL is required, and WebLogic Server is the client, the following system properties must be applied to the `weblogic.properties` file before you upgrade:

- `-Dweblogic.security.SSL.AllowAnonymousCipher=true`
- `-Dweblogic.security.SSL.ignoreHostnameVerification=true`

 **Note:**

Setting these properties can make the WebLogic Server susceptible to MITM attacks, since anonymous cipher suites are enabled, and the client connections are without Hostname Verification checking.

Oracle strongly recommends that you to use either server or client/server SSL authentication when using OID with WebLogic Server 12c.

Removing the Server Instance Scope from OWSM Policy Sets

The Server Instance Scope in policy sets was not recommended in 11g (11.1.1.7.0) and is not supported in 12c. However, if you have policy sets that use the Server Instance Scope, they are disabled during the upgrade to 12c. Therefore, you must

remove the Server Instance Scope from all the 11g policy sets before upgrading to 12c.

For instructions, see [Editing a Policy Set](#) in *Security and Administrator's Guide for Web Services* in the Oracle Fusion Middleware 11g Release 1 (11.1.1.7.0) documentation library.

Cloning Predefined Documents and Migrating OWSM Policy Attachments

When upgrading, it is important to note that any predefined documents that have not been customized for your environment are replaced with read-only versions, and new, predefined, read-only documents are added. However, any existing predefined documents that have been customized and any user-created custom policies in the repository are not overwritten.

To ensure that you always get all of the latest policies, Oracle recommends that you clone any predefined documents that you have modified and migrate any policy attachments. For details, see *Upgrading the OWSM Repository in Securing Web Services and Managing Policies with Oracle Web Services Manager*.

Pre-Upgrade Checklist

The Pre-Upgrade Checklist identifies tasks that can be performed before you begin your upgrade to ensure you have a successful upgrade and limited downtime.

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

Note:

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

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

Task	Description
<p>Required Create a complete backup of your existing environment.</p>	<p>Back up all system-critical files and database(s) that contain any schemas that are to be upgraded. If the upgrade fails, you must restore your pre-upgrade environment and begin the upgrade again.</p> <p>See Creating a Complete Backup.</p> <ul style="list-style-type: none"> • Make sure that your backup includes the schema version registry table. See Backing Up the Schema Version Registry Table. • If you modified any of the startup scripts in your existing domain, you will need to copy them to temporary directory location (outside of the existing domain) during the upgrade and redeploy them after the upgrade.
<p>Optional 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 Cloning Your Production Environment.</p>
<p>Required Verify that you are installing and upgrading your product on a supported hardware and software configuration. 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 documents. 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>Make sure that you have applied the latest patches to your components before you upgrade.</p> <p>See Verifying Certification and System Requirements.</p>
<p>Required for 32-bit Operating Systems Only Migrate to a 64-bit operating system before you can upgrade.</p>	<p>This is required only if you are currently running an unsupported 32-bit operating system.</p>
<p>Optional 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 Updating Policy Files when Using Enhanced Encryption (AES 256).</p>
<p>Optional 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 Purging Unused Data.</p>
<p>Required for Oracle Database Users Only Before upgrading an Edition-Based Redefinition (EBR) enabled schema, you must connect to the database server and create an edition on the database server for 12c (12.2.1.3.0).</p>	<p>If you are using an Edition-Based Redefinition (EBR) database, you must create the edition before starting the upgrade.</p> <p>See Creating an Edition on the Server for Edition-Based Redefinition.</p>

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

Task	Description
Optional Create a Non-SYSDBA user to run the Upgrade Assistant.	Oracle recommends that you create the FMW user to run Upgrade Assistant. User FMW can run the Upgrade Assistant without system administration privileges. See Creating a Non-SYSDBA User to Run the Upgrade Assistant
Optional Identify which schemas are currently in your domain before you begin.	It is important that you know which schemas are in your pre-upgrade domain before you start the upgrade. You should know the schema owner names and passwords, as well as the versions of each schema.

Creating a Complete Backup

Before you start an upgrade, back up all system-critical files, including the 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.

The Upgrade Assistant Prerequisites screen prompts you to acknowledge that backups have been performed before you proceed with the actual upgrade. However, note that 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 \(12.2.1.3.0\) in *Planning an Upgrade of Oracle Fusion Middleware*](#)
- [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 claspath
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 `EXISTING_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*.

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.

 **WARNING:**

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

- [Verify Your Environment Meets Certification Requirements](#)
Oracle has tested and verified the performance of your product on all certified systems and environments. Make sure that you are installing your product on a supported hardware or 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 (12.2.1.3.0).
- [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 or 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 Technology Network. 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, the System Requirements and Specifications document should be used to 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 Technology Network (OTN).



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 12c (12.2.1.2.0) software is working properly on the 64-bit machine, and only then perform the upgrade to Oracle Fusion Middleware 12c (12.2.1.3.0).

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 12c (12.2.1.2.0) 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 12c \(12.2.1.2.0\) 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 12c \(12.2.1.2.0\) 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 12c (12.2.1.2.0). The directory structure on the target machine must be identical to the structure of the host machine.
- [Install the 12c \(12.2.1.3.0\) Product Distributions on the Target Machine](#)
Oracle recommends an Out-of-Place approach for upgrade. Therefore, you must install the 12c (12.2.1.3.0) 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.

Stop the Managed Servers

To stop a WebLogic Server Managed Server, use the `stopManagedWebLogic` script:

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

When prompted, enter your user name and password.

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

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

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 12c (12.2.1.2.0) 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:

- `12c_DOMAIN_HOME`

- `12c/nodemanager` directory located in `12c_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 12c (12.2.1.2.0) 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 12c (12.2.1.2.0) 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 12c (12.2.1.2.0). 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 (12.2.1.3.0) Product Distributions on the Target Machine

Oracle recommends an Out-of-Place approach for upgrade. Therefore, you must install the 12c (12.2.1.3.0) 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 12c (12.2.1.2.0) Node Manger 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 (12.2.1.3.0).

Review the Fusion Middleware database requirements before starting the upgrade to ensure that the database hosting Oracle Fusion Middleware is supported and has sufficient space to perform an upgrade. See the Certification Matrix for 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 (12.2.1.3.0)* 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 Technology Network (OTN) to verify that the JDK you are using is supported.

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

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

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

For more information on the difference between generic and platform-specific installers, see *Understanding the Difference Between Generic and Platform-Specific Distributions* in the *Oracle Fusion Middleware Download, Installation, and Configuration Readme Files*.

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.

Creating an Edition on the Server for Edition-Based Redefinition

Before upgrading an Edition-Based Redefinition (EBR) enabled schema, you must connect to the database server and create an edition on the database server for 12c (12.2.1.3.0).

Edition-based redefinition enables you to upgrade an application's database objects while the application is in use, thus minimizing or eliminating downtime. This is accomplished by changing (redefining) database objects in a private environment known as an edition. Only when all the changes have been made and tested, you make the new version of the application available to users.

**Note:**

This task must be completed by an Oracle Database User with DBA privileges.

Before upgrading an Edition-Based Redefinition (EBR) enabled schema, you must connect to the database server and create an edition on the database server for 12c (12.2.1.3.0). The new edition for 12c (12.2.1.2.0) must be a child of your existing 12c or 12c (12.2.1.3.0) edition.

To create an edition on the database server, sign in as an SYS user (or another Oracle user that has DBA privileges) and enter the following command:

```
create edition Oracle_FMW_12_2_1_4_0 as child of  
Oracle_FMW_12_2_1_3_0;
```

where Oracle_FMW_12_2_1_3_0 is an example of the edition name you specified in RCU 12.2.1.3 when the 12.2.1.3 schemas were created. Be sure to provide the actual name used when creating the edition.

The following message notifies you that the edition is created successfully:

```
Edition created.
```

During the upgrade, you are prompted to launch the Reconfiguration Wizard to reconfigure your existing domain. Before running the Reconfiguration Wizard, you must specify the database default edition. Use the following SQL command to manually set up the default edition name for the database, for example:

```
ALTER DATABASE DEFAULT EDITION = Oracle_FMW_12_2_1_4_0;
```

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\$*xatrans* 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\$*xatrans* 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\$*xatrans* table, then remove the following line from the script:

```
grant select on v$xatrans 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_TABLESPACE_USAGE_METRICS to FMW with grant option;
grant select on SYS.DBA_DATA_FILES to FMW with grant option;
grant select on SYS.V_$ASM_DISKGROUP to FMW with grant option;
grant select on v$xatrans to FMW with grant option;
```

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

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

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

3

Upgrading Oracle Fusion Middleware Infrastructure from 11g Release

You can upgrade Oracle Fusion Middleware Infrastructure from an earlier Oracle Fusion Middleware Release 11g version to the latest 12c (12.2.1.3.0) release version.

Oracle Fusion Middleware Upgrade Assistant supports an out-of-place upgrade to 12c (12.2.1.3.0) release for the following Oracle Fusion Middleware Infrastructure 11g releases:

- Oracle Fusion Middleware Infrastructure FMW 11g Release 1, Version: 11.1.1.6.0
- Oracle Fusion Middleware Infrastructure FMW 11g Release 1, Version: 11.1.1.7.0
- Oracle Fusion Middleware Infrastructure FMW 11g Release 1, Version: 11.1.1.9.0

The following topics describe how to upgrade Oracle Fusion Middleware Infrastructure to 12c (12.2.1.3.0) release:

- [About the Oracle Fusion Middleware Infrastructure Upgrade Process \(11g to 12c\)](#)
Review the flowchart and roadmap for an overview of the upgrade process for Oracle Fusion Middleware Infrastructure from 11g to 12c.
- [Installing Oracle Fusion Middleware Infrastructure](#)
Installing Fusion Middleware Infrastructure creates an Oracle home directory and lays down supporting software to install other Fusion Middleware products.
- [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.
- [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 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.
- [Creating the Required 12c Schemas with the RCU](#)
When upgrading, you must create the required schemas. You can use the Repository Creation Utility (RCU) to create customized schemas or, optionally, you can use the Upgrade Assistant to create schemas using the default schema settings. 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.
- [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.
- [Starting Servers and Processes](#)
After a successful upgrade, restart all processes and servers, including the Administration Server and any Managed Servers.
- [Using the Upgrade Validation Checklist](#)
After the upgrade, make sure that you can successfully complete the basic administration tasks, such as verifying whether you are able to start the Node Manager, Administration Server, Webtier, Administration Console, and Enterprise Manager Fusion Middleware Control.
- [Reapplying Custom Configuration Settings to setDomainEnv](#)
To complete the upgrade of your application environment to 12c it might be necessary to reapply any custom configuration settings to startup scripts, such as `setDomainEnv`. During the upgrade, the scripts are overwritten with new 12c versions. You must manually reapply the custom configuration settings you had made in previous releases.
- [Configuring an Oracle Fusion Middleware 12c Audit Data Store](#)
If you were using a file-based audit store in your 11g deployment, then after the upgrade to Oracle Fusion Middleware 12c, you should enable the loading of audit data to a database-based Audit Data Store.
- [Maintaining the Security Status of Older Java EE Web Service Applications](#)
The introduction of global policy attachment support for Java EE web services and clients in 12c may impact the backwards compatibility of existing Java EE web services and clients (12.1.2 and earlier). If a Java EE web service or client endpoint that depends on the absence of a policy falls within the scope of a global policy attachment, the presence of the globally-attached policy can alter the security behavior of that endpoint.
- [Documentation Resources for Managing your Oracle Fusion Middleware 12c Software](#)
This topic provides a list of common administration tasks you likely want to perform after upgrading to Infrastructure 12c and associated documentation resources.
- [Using Your 11g Application Deployments in Oracle Fusion Middleware 12c](#)
After you upgrade to Oracle Fusion Middleware 12c, the custom Java and Application Development Framework (ADF) you previously deployed on Oracle Fusion Middleware 11g work as they did in Oracle Fusion Middleware 11g. However, there are some new features and capabilities available in ADF 12c and in JDeveloper 12c.
- [If Your Existing Environment is a Clustered Configuration...](#)
If your existing environment is a clustered configuration, then you must apply the changes to other cluster members in the domain by using the pack and unpack utilities.
- [Starting Servers and Processes](#)
After a successful upgrade, restart all processes and servers, including the Administration Server and any Managed Servers.

About the Oracle Fusion Middleware Infrastructure Upgrade Process (11g to 12c)

Review the flowchart and roadmap for an overview of the upgrade process for Oracle Fusion Middleware Infrastructure from 11g to 12c.

Figure 3-1 Upgrade Process Flowchart for Oracle Fusion Middleware Infrastructure from 11g to 12c

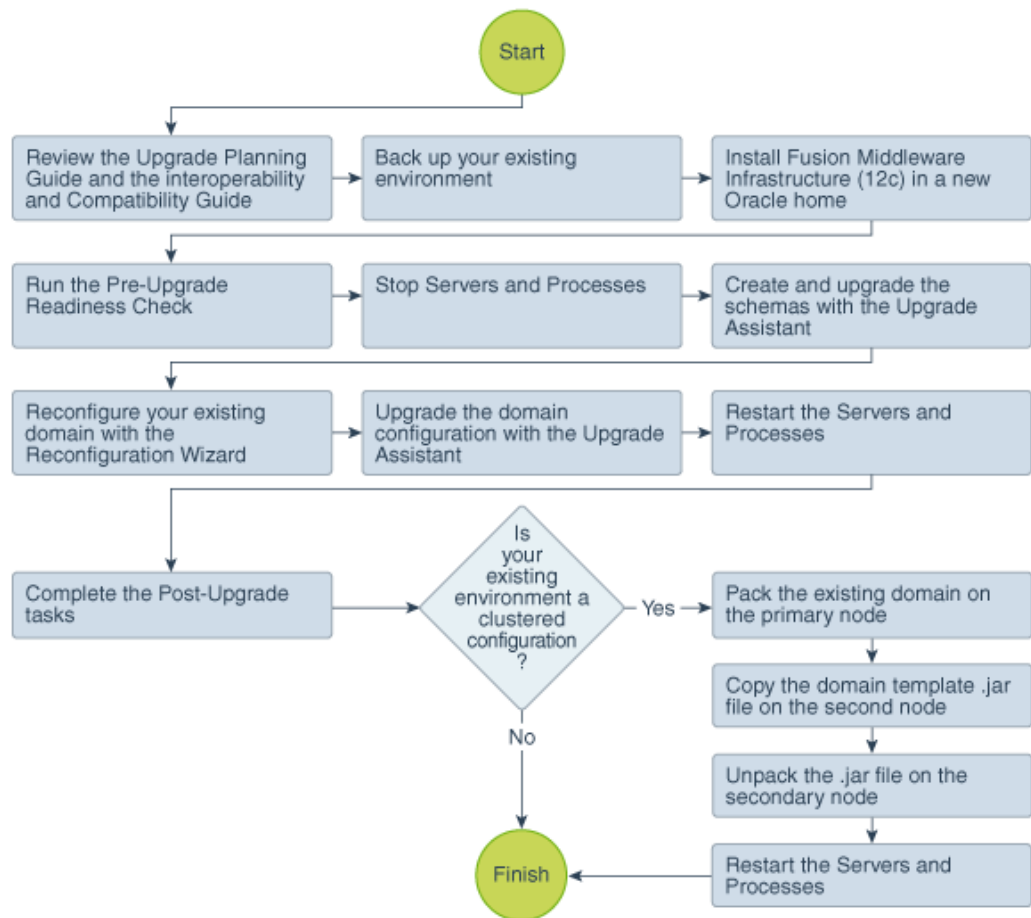


Table 3-1 lists the high-level steps that you need to perform to upgrade to Oracle Fusion Middleware Infrastructure Release 12.2.1.3.0:

Table 3-1 Tasks for Upgrading Oracle Fusion Middleware Infrastructure from a 11g Release

Task	Description
<p>Optional Learn about the interoperability and compatibility factors that could affect how you upgrade to Oracle Fusion Middleware Infrastructure 12.2.1.3.0.</p>	<p>It is important to understand how two or more Oracle Fusion Middleware products of the same version or different versions work together (interoperate) in a supported Oracle Fusion Middleware configuration.</p> <p>You can learn more about interoperability and compatibility in Oracle® Fusion Middleware Understanding Interoperability and Compatibility.</p>
<p>Required If you have not done so already, review the introductory topics in this guide and complete the required pre-upgrade tasks.</p>	<p>The pre-upgrade tasks include cloning your production environment, verifying system requirements and certifications, purging unused data, and creating non-SYSDBA user.</p> <p>For a complete list of pre-upgrade tasks, see Pre-Upgrade Checklist.</p>
<p>Required Download and install the 12.2.1.3.0 Fusion Middleware Infrastructure distribution.</p>	<p>The Infrastructure distribution combines the WebLogic Server and the Java Required Files (JRF) that are required to set up the foundation to install other Fusion Middleware products.</p> <p>As per the upgrade topology defined in this guide, you must install the Infrastructure in a new Oracle home. Therefore, follow the procedure described in Installing Oracle Fusion Middleware Infrastructure.</p>
<p>Optional Run the Readiness Check.</p>	<p>See Running a Pre-Upgrade Readiness Check.</p>
<p>Required Stop the servers and processes.</p>	<p>Before starting the upgrade process, stop all the servers, components, and processes.</p> <p>See Stopping Servers and Processes.</p>
<p>Required Create and upgrade the required 12c schemas.</p>	<p>The Upgrade Assistant allows you to upgrade and/or create the required schemas for your domain. See Creating and Upgrading Schemas Using the Upgrade Assistant. The Upgrade Assistant creates schemas with the default tablespace settings. If you want to specify custom tablespaces for your schemas, then you must create the required schemas with the RCU. See Creating the Required 12c Schemas with the RCU.</p>
<p>Required Reconfigure the existing domain with the Reconfiguration Wizard.</p>	<p>When you run the Reconfiguration Wizard on your existing domain, it prepares your domain for upgrade by selecting and applying the reconfiguration templates. It also tests the JDBC data sources and component schemas that are present within your domain.</p> <p>To reconfigure you domain, follow the procedure described in About Reconfiguring the Domain.</p>
<p>Required Upgrade the existing domain configurations with the Upgrade Assistant.</p>	<p>After you have reconfigured your existing domain, you must run the Upgrade Assistant to upgrade all configurations used by your existing domain.</p> <p>You can see all the components within your domain that will be upgraded on the Component List screen when you run the Upgrade Assistant. For the complete procedure, see Upgrading Domain Component Configurations.</p>

Table 3-1 (Cont.) Tasks for Upgrading Oracle Fusion Middleware Infrastructure from a 11g Release

Task	Description
Required Restart the servers and processes.	The upgrade process is complete. You can now restart the servers, components, and processes. See Starting Servers and Processes .
Required Perform the post-upgrade tasks.	For a list of post-upgrade tasks, see Using the Upgrade Validation Checklist .
Required only if your existing environment is a clustered configuration Pack the existing domain on the primary node.	Run the <code>pack.sh cmd</code> script to pack your existing domain in a <code>domaintemplate.jar</code> file on the primary node. See If Your Existing Environment is a Clustered Configuration...
Required only if your existing environment is a clustered configuration Copy the domain template.jar file on the secondary node.	Copy the <code>domaintemplate.jar</code> file on the secondary node so that you can unpack the contents of the file on the secondary node. See If Your Existing Environment is a Clustered Configuration...
Required only if your existing environment is a clustered configuration Unpack the jar file on the secondary node	Run the <code>unpack.sh cmd</code> script to unpack the <code>domaintemplate.jar</code> file on the secondary node. See If Your Existing Environment is a Clustered Configuration...
Required only if your existing environment is a clustered configuration Restart the servers and processes.	The upgrade process is complete. You can now restart the servers, components, and processes. See Starting Servers and Processes .

Installing Oracle Fusion Middleware Infrastructure

Installing Fusion Middleware Infrastructure creates an Oracle home directory and lays down supporting software to install other Fusion Middleware products.

The supported JDK version for 12c (12.2.1.3.0) is 1.8.0_131. Make sure that you upgrade to the latest JDK version before installing the 12c (12.2.1.3.0) software.

Note:

If you are upgrading from a previous 12c release (12.1.3.0, 12.2.1.0 or 12.2.1.1), then you must install the 12c (12.2.1.3.0) distributions into a new Oracle home. Do not attempt to reuse the existing Oracle home for this upgrade. Upgrading to 12c (12.2.1.3.0) is not a patch release.

To install Oracle Fusion Middleware Infrastructure distribution:

1. Sign in to the target system.
2. Download the Oracle Fusion Middleware Infrastructure (`fmw_12.2.1.3.0_infrastructure_generic.jar`) from [Oracle Technology Network](#) or [Oracle Software Delivery Cloud](#) to your target system.
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) `JDK_HOME/bin/java -jar fmw_12.2.1.3.0_infrastructure_generic.jar`
 - (Windows) `JDK_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 *Understanding Directories for Installation and Configuration in Oracle Fusion Middleware Planning an Installation of Oracle Fusion Middleware*.
9. On the Installation Type screen, select **Fusion Middleware Infrastructure** and 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 Security Updates screen, enter your My Oracle Support account information so you can receive the latest product information and security updates via your My Oracle Support account.

This screen appears the first time you install an Oracle product on a host.

If you do not have an Oracle Support account and you are sure that you want to skip this step, clear the check box and verify your selection in the follow-up dialog box.

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

If you want to save these options to a response file, click **Save Response File** and enter the response file location and name. The response file collects and stores all the information that you have entered, and enables you to perform a silent installation (from the command line) at a later time.

Click **Install** to begin the installation.

13. On the Installation Progress screen, when the progress bar displays 100%, click **Finish** to dismiss the installer, or click **Next** to see a summary.
14. The Installation Complete screen displays the Installation Location and the Feature Sets that are installed. Review this information and click **Finish** to close the installer.

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) `NEW_ORACLE_HOME/oracle_common/upgrade/bin`
 - (Windows) `NEW_ORACLE_HOME\oracle_common\upgrade\bin`
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, set the `DISPLAY` environment variable to the system name or IP address of your local workstation, and rerun Upgrade Assistant.

If you continue to receive these errors after setting `DISPLAY`, try launching another GUI tool, such as `vncconfig`. If you see the same errors, your `DISPLAY` environment variable may still not be set correctly.

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

- [Upgrade Assistant Parameters](#)

Upgrade Assistant Parameters

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

Table 3-2 Upgrade Assistant Command-Line Parameters

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

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

Parameter	Required or Optional	Description
<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>NEW_ORACLE_HOME/ oracle_common/upgrade/ logs NEW_ORACLE_HOME/ oracle_common/upgrade/ temp</pre> <p>(Windows)</p> <pre>NEW_ORACLE_HOME\oracle_c ommon\upgrade\logs NEW_ORACLE_HOME\oracle_c ommon\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 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 March 30 11:15:52 EDT
2019
```

```
Log file is located at: ORACLE_HOME/oracle_common/upgrade/logs/
ua2016-05-30-11-14-06AM.log
```

```
Readiness Check Report File: 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 12.2.1.4.0.
```

```
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 12c Enterprise
Edition Release 12.2.1.4.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing
options
  Completed schema test: TEST_DATABASE_VERSION --> Test that the
database server version number is supported for upgrade +++ PASS
  Finished readiness check of Oracle Metadata Services with status:
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:

```

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_INDEXES --> 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 (12.2.1.3.0) by using the RCU.

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 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.

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 system components, such as Oracle HTTP Server, use the `stopComponent` script:

- (UNIX) `EXISTING_DOMAIN_HOME/bin/stopComponent.sh component_name`
- (Windows) `EXISTING_DOMAIN_HOME\bin\stopComponent.cmd component_name`

You can stop system components in any order.

Step 2: Stop the Managed Servers

To stop a WebLogic Server Managed Server, use the `stopManagedWebLogic` script:

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

When prompted, enter your user name and password.

Step 3: Stop Oracle Identity Management Components

Stop any Oracle Identity Management components, such as Oracle Internet Directory:

- (UNIX) `EXISTING_DOMAIN_HOME/bin/stopComponent.sh component_name`
- (Windows) `EXISTING_DOMAIN_HOME\bin\stopComponent.cmd component_name`

Step 4: 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) `EXISTING_DOMAIN_HOME/bin/stopWebLogic.sh`
- (Windows) `EXISTING_DOMAIN_HOME\bin\stopWebLogic.cmd`

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

Step 5: Stop Node Manager

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

Alternatively, after setting 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*.

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.

- [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.

- [Creating and Upgrading 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.

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.

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

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

- [Upgrade Assistant Parameters](#)

Upgrade Assistant Parameters

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

Table 3-4 Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
-readiness	Required for readiness checks Note: Readiness checks cannot be performed on standalone installations (those not managed by the WebLogic Server).	Performs the upgrade readiness check without performing an actual upgrade. Schemas and configurations are checked. Do not use this parameter if you have specified the -examine parameter.
-threads	Optional	Identifies the number of threads available for concurrent schema upgrades or readiness checks of the schemas. The value must be a positive integer in the range 1 to 8. The default is 4.
-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"> • TRACE • NOTIFICATION • WARNING • ERROR • INCIDENT_ERROR The default logging level is NOTIFICATION. Consider setting the -logLevel TRACE attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if -logLevel TRACE is used.

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

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

Creating and Upgrading Schemas Using the Upgrade Assistant

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

To upgrade product schemas with the Upgrade Assistant:

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

Note:

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

2. On the Selected Schemas screen, select **All Schemas Used by a Domain**.
 - **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.

Click **Next**.

3. If you selected **All Schemas Used by a Domain**: On the Component List screen you will see two lists of schemas. The first list shows the components whose schemas are present in the domain and will be upgraded. The second list shows the list of missing schemas that may need to be created. If none of the required schemas are missing, you will only see the first list. Review both lists and click **Next**.

The Upgrade Assistant will attempt to create any missing schemas using the schema credentials used to create the existing domain schemas. You do not to launch the Repository Creation Utility.

If you want to exclude some components or schemas from the list, navigate back to the All Schemas screen and select **Individually Selected Schemas**. This option allows you to select only those schemas you want included in the upgrade.

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 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.

6. On the Create Schemas screen, indicate whether you want the Upgrade Assistant to create the missing schemas for the domain. You cannot perform an upgrade if any required schemas are missing.

The **Create missing schemas for the specified domain** option is enabled by default. The Upgrade Assistant will attempt to create the missing schemas for the domain using the database connection details and schema owner name provided.

If the same password is used for all schemas select **Use same passwords for all schemas**. Enter and confirm the password in the table. You only have to specify the password once.

If you do not want the Upgrade Assistant to create these schemas for you, deselect this option and click **Next**. You will have to run the Repository Creation Utility (RCU) to create the schemas.

7. The Create Schemas Default screen appears if you have chosen to create the schemas in this session of the Upgrade Assistant.

 **Note:**

The schemas are created with the default tablespace settings. If you want to customize the tablespace settings for your schemas, then create the required schemas with the RCU.

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

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

 **Note:**

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

9. On the Upgrade Summary screen, review the summary of the schemas that will be upgraded and/or created.

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 **Next**.

10. The Create Schemas Progress screen displays the progress and the status of the schemas that are created. When complete, click **Upgrade**.
11. 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**.

12. 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 `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 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. Some schemas do not require an upgrade to this release and will retain their pre-upgrade version number.

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

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

Creating the Required 12c Schemas with the RCU

When upgrading, you must create the required schemas. You can use the Repository Creation Utility (RCU) to create customized schemas or, optionally, you can use the Upgrade Assistant to create schemas using the default schema settings. 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:**

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. If you are upgrading from 12c, 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 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 12c 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 12c. 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 schemas with the RCU:

1. (Optional) If you are upgrading from 12c, 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 12c schemas. See the pertinent table below.

Table 3-5 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: examplehost.exampledomain.com For Oracle RAC databases, specify the VIP 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: examplehost.exampledomain.com
Username	Enter the user name for your database. The default user name is SYS.
Password	Enter the password for your database user.
Role	Select the database user's role from the drop-down list: Normal or SYSDBA

Table 3-6 Connection Credentials for MySQL Databases

Option	Description and Example
Host Name	Specify the host name, IP address, or complete server name in <code>host\server</code> format of the server where your database is running.
Port	Specify the port number for your database.
Database Name	Specify the name of your database.

Table 3-6 (Cont.) Connection Credentials for MySQL Databases

Option	Description and Example
Username	Specify the name of a user with administrator privileges.
Password	Enter the password for your database user.

Table 3-7 Connection Credentials for Microsoft SQL Server Databases

Option	Description and Example
Unicode Support	Select Yes or No from the drop-down list.
Server Name	Specify the host name, IP address, or complete server name in <i>host\server</i> format of the server where your database is running. MSSQL named instances: A named instance is identified by the network name of the computer and the instance name that you specify during installation. The client must specify both the server name and the instance name when connecting.
Port	Specify the port number for your database.
Database Name	Specify the name of your database.
Username	Specify the name of a user with administrator privileges.
Password	Enter the password for your database user.

Table 3-8 Connection Credentials for IBM DB2 Databases

Option	Description and Example
Server Name	Specify the host name, IP address, or complete server name in <i>host\server</i> format of the server where your database is running.
Port	Specify the port number for your database.
Database Name	Specify the name of your database.
Username	Specify the name of a user with DB Owner privileges. The default user name for IBM DB2 databases is <code>db2inst1</code> .
Password	Enter the password for your database user.

If the prerequisite check is successful, click **OK** to continue to the next screen. If the check fails, review the details you entered and try again.

- 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 **AS Common Schemas**.

 **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.

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 Management (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/init-info/domain-info.xml` that are similar to this example:

```
<!--extension-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/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 Domain with the Reconfiguration Wizard](#)
The Reconfiguration Wizard reconfigures the domain while retaining the location of the 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 domain directory:

1. Copy the source domain to a separate location to preserve the contents.
(Windows) `copy C:\domains\mydomain to C:\domains\mydomain_backup.`
(UNIX) `cp mydomain /domains/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 collocated managed servers before starting the reconfiguration process. See [Stopping Servers and Processes](#) .

To start the Reconfiguration Wizard in graphical mode:

1. Sign in to the system on which the domain resides.
2. Open the command shell (on UNIX operating systems) or open a command prompt window (on Windows operating systems).
3. **Edition Based Database Users Only:** If your schemas are configured with EBR database, a default edition name must be manually supplied before you run the Reconfiguration Wizard.

Run the following SQL command to set the default edition:

```
ALTER DATABASE DEFAULT EDITION = edition_name;
```

where *edition_name* is the child edition name.

4. Go to the `oracle_common/common/bin` directory:
 - (UNIX) `NEW_ORACLE_HOME/oracle_common/common/bin`
 - (Windows) `NEW_ORACLE_HOME\oracle_common\commom\bin`
5. 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.

 **Note:**

When you run this command, the following error message might appear to indicate that the default cache directory is not valid:

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

You can change the cache directory by setting the environment variable `CONFIG_JVM_ARGS`. For example:

```
CONFIG_JVM_ARGS=-Dpython.cachedir=valid_directory
```

Reconfiguring the Domain with the Reconfiguration Wizard

The Reconfiguration Wizard reconfigures the domain while retaining the location of the domain. Navigate through the screens in the Reconfiguration Wizard to reconfigure your existing domain.

Important:

If the source is a clustered environment, run the Reconfiguration Wizard on the primary node only. Use the pack/unpack utility to apply the changes to other cluster members in the domain as described in [If Your Existing Environment is a Clustered Configuration...](#)

To reconfigure the domain:

1. Sign in to the system on which the domain resides.
2. **Edition Based Database Users Only:** If you have configured your schemas with Edition-Based Reassociation, you must manually supply a default edition name before running the Reconfiguration Wizard.

To set the default edition, enter the following SQL command:

```
ALTER DATABASE DEFAULT EDITION = edition_name;
```

Where, *edition_name* is the name of the default database edition.

3. 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**.
4. 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.
5. 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.

6. On the JDBC Data Sources screen, configure the JDBC data sources defined in your domain source.

The JDBC data sources associated with the products for which you are creating the domain are listed in the lower half of the screen. A JDBC data source contains a pool of database connections that are created when the data source instance is created, deployed or targeted, or at server startup. Applications look up a data source on the JNDI tree, and then request a connection. When the applications no longer need the connections, they return the connections to the connection pool in the data source.

From the **Data Source Name** drop-down list, select the data source(s) for which you want to specify the settings. The values that you specify are displayed in the appropriate columns in the data source list, for the selected data source.

For Oracle RAC Configuration for data sources, you can select one of the three options:

- Convert to GridLink
- Convert to RAC multi data source
- Don't convert

For more information about each option, click **Help**.

After specifying the details, click **Next**.

If you do not select any data sources on the JDBC Data Sources screen, the following warning displays:

Missing Driver

Click **Ok** to proceed without verification, click **Cancel** to return to the JDBC Data Sources page.

In this case, if you click **Ok**, the data sources are not verified.

7. On the JDBC Data Sources Test screen, select the check box for the data source connection you configured on the JDBC Data Sources screen and click **Test Selected Connections** to test the data source connection.

 **Note:**

To test the database connections, the database to which you are connecting must be running. If you do not want to test the connections at this time, do not select any data sources. Click **Next** to continue.

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

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

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

 **Note:**

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

 **Note:**

For any existing 12c (12.2.1.2.0) datasource, the reconfiguration will preserve the existing values. For new datasources where the schema was created for 12c (12.2.1.3.0) by the RCU, the default connection data will be retrieved from the `_STB` schema. If no connection data for a given schema is found in the `_STB` schema, then the default connection data is used.

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

 **Note:**

If you are upgrading from 12c (12.2.1.2.0), and your database has `_OPSS` or `_IAU` 12c (12.2.1.2.0) database schemas, you must manually enter database connection details for those schemas. These schemas were not required in 12c and had to be created manually. Users could assign any name to these schemas, therefore the Reconfiguration Wizard does not recognize them. When providing connection information for `_IAU`, use the `IAU_APPEND` user information.

9. On the JDBC Component Schema Test screen, select all the component schemas and click **Test Selected Connections** to test the connection for each schema. The result of the test is indicated in the Status column.

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

10. The Node Manager screen is only displayed if the domain you are reconfiguring is currently using a per host Node Manager.

On the Node Manager screen, select the Node Manager configuration to use for the reconfigured domain. The resulting configuration depends on the combination of options you select for **Node Manager Type** and **Node Manager Configuration**.

Table 3-9 Field Descriptions for Node Manager Screen

Option	Description
Per Domain Default Location	If you select this option, the Node Manager home is redefined to <code>\$domain_name/nodemanager</code> and you cannot edit the Node Manager home.

Table 3-9 (Cont.) Field Descriptions for Node Manager Screen

Option	Description
Per Domain Custom Location	Select this option if you want the per domain Node Manager configuration files to be created in a specific location for this domain. Specify the directory in the Node Manager Home field, or click Browse to use the navigation tree to select the location. The specified directory must be empty. The <code>nodemanager.properties</code> and <code>nodemanager.domains</code> files are created in this directory.
Node Manager Home	If you selected the Per Domain Custom Location option, click Browse to navigate to the directory location that you want to use to store the per domain Node Manager configuration.
Manual Node Manager Setup	If you select this option, creation of the Node Manager configuration for the domain is skipped (all remaining fields cannot be modified), and if you want to use Node Manager in the domain, you must manually configure Node Manager as described in <i>Completing the Node Manager Configuration</i> . The reconfigured domain will still use a per host Node Manager configuration. You should also select this option if your existing domain is not configured to use Node Manager and you do not want to use Node Manager in the reconfigured domain. For more information about Node Manager configuration, see <i>Administering Node Manager for Oracle WebLogic Server</i> .
Node Manager Configuration	Select one of the following two options. These fields are not available if you selected Manual Node Manager Setup .
Create New Configuration	A per domain Node Manager configuration will be automatically created for the reconfigured domain using default settings in <code>nodemanager.properties</code> . If necessary, you can modify <code>nodemanager.properties</code> after the domain has been successfully reconfigured.
Migrate Existing Configuration	The existing per host Node Manager configuration will be migrated to a per domain configuration for the reconfigured domain. This does not include environment-specific settings for <code>ListenAddress</code> , <code>ListenPort</code> , <code>StartScriptName</code> , <code>JavaHome</code> , and <code>LogFile</code> .
Node Manager Home	If you selected the Migrate Existing Configuration option, enter or browse to the Node Manager home directory that you want to migrate to the reconfigured domain.

Table 3-9 (Cont.) Field Descriptions for Node Manager Screen

Option	Description
Apply Oracle Recommended Defaults	<p>If you selected the Migrate Existing Configuration option, select this check box if you want to use Oracle-recommended defaults in the <code>nodemanager.properties</code> file. Deselect this check box if you want to continue using the settings in the <code>nodemanager.properties</code> file being migrated.</p> <p>Oracle-recommended properties with default values are as follows:</p> <pre> LogLimit=0 AuthenticationEnabled=true LogLevel=INFO DomainsFileEnabled=true NativeVersionEnabled=true LogToStderr=true SecureListener=true LogCount=1 StopScriptEnabled=false QuitEnabled=false LogAppend=true StateCheckInterval=500 CrashRecoveryEnabled=false StartScriptEnabled=true LogFormatter=weblogic.nodemanager.server.LogFormatter ListenBacklog=50 </pre>
Node Manager Credentials: Username, Password	Specify the username and password that you want to use to start Node Manager in the reconfigured domain.

- 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**.

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

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

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

 **Note:**

The location of the domain does not change when you reconfigure it.

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

During this process:

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

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

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

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

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

For all the Advanced Configurations-related screens, such as Managed Servers, Clusters, Machines, HTTP Proxy Applications, Coherence Clusters, and System Components, see *Upgrading Oracle WebLogic Server*.

If an error occurs while reconfiguring your domain, refer to the Important Notes About the Domain Upgrade Process in *Upgrading Oracle WebLogic Server*.

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 the Domain Configurations with the Upgrade Assistant](#)
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.

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

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

- [Upgrade Assistant Parameters](#)

Upgrade Assistant Parameters

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

Table 3-10 Upgrade Assistant Command-Line Parameters

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

Table 3-10 (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 <code>-readiness</code> parameter.
-logLevel <i>attribute</i>	Optional	Sets the logging level, specifying one of the following attributes: <ul style="list-style-type: none">• TRACE• NOTIFICATION• WARNING• ERROR• INCIDENT_ERROR The default logging level is NOTIFICATION. 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.

Table 3-10 (Cont.) Upgrade Assistant Command-Line Parameters

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

Upgrading the Domain Configurations with the Upgrade Assistant

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 All Configurations screen, select **All Configurations Used by a Domain** and specify your domain location in the **Domain Directory** field by entering it directly or by clicking **Browse** to use a navigation tree to select a valid 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 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.

7. 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.

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 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**.

9. 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 Oracle Enterprise Manager Fusion Middleware Control Console, go to:
`http://administration_server_host:administration_server_port/em`

 **Note:**

After upgrade, make sure you run the administration tools from the new 12c (12.2.1.3.0) Oracle home directory and not from the previous Oracle home directory.

During the upgrade process, some OWSM documents, including policy sets and predefined documents such as policies and assertion templates, may need to be upgraded. If a policy set or a predefined document is upgraded, its version number is incremented by 1.

If you created the FMW user to run the Upgrade Assistant, ensure that you delete the account after verifying your upgrade was successful.

Starting Servers and Processes

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

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

 **Note:**

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

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

Step 1: Start the Administration Server

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

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

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

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

Step 2: Start Node Manager

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

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

Step 3: Start Oracle Identity Management Components

Start any Oracle Identity Management components, such as Oracle Internet Directory, that form part of your environment:

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

Step 4: Start the Managed Servers

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

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

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.

Step 5: Start System Components

To start system components, such as Oracle HTTP Server, use the `startComponent` script:

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

You can start system components in any order.

Using the Upgrade Validation Checklist

After the upgrade, make sure that you can successfully complete the basic administration tasks, such as verifying whether you are able to start the Node Manager, Administration Server, Webtier, Administration Console, and Enterprise Manager Fusion Middleware Control.

Note:

The order in which you start the following servers is important and failure to start (or stop) them in the correct order can cause issues with the deployment.

1. Verify that you are able to start the Node Managers.
2. Verify that you are able to start the Administration Server and any Managed Servers (if included) from the original Domain Home `bin` directory. Windows operating system users may find it useful to start the servers from a new command prompt (and not the one used to launch the 12c Upgrade Assistant).

 **Note:**

OHS does not need a Managed Server for its own configuration.

3. Verify that you are able to start the Webtier (OHS server).
4. Verify that you are able to access the Administration console and Enterprise Manager using the following URLs:

Administration console: `http://
machinename.my_company_com:administration_port/console`

Enterprise Manager: `http://
machinename.my_company_com:administration_port/em`

Reapplying Custom Configuration Settings to setDomainEnv

To complete the upgrade of your application environment to 12c it might be necessary to reapply any custom configuration settings to startup scripts, such as `setDomainEnv`. During the upgrade, the scripts are overwritten with new 12c versions. You must manually reapply the custom configuration settings you had made in previous releases.

See [Re-apply Customizations to Startup Scripts](#).

 **Note:**

To prevent losing your custom configuration settings in a future upgrade, see [Maintaining Your Custom setDomainEnv Settings](#).

Configuring an Oracle Fusion Middleware 12c Audit Data Store

If you were using a file-based audit store in your 11g deployment, then after the upgrade to Oracle Fusion Middleware 12c, you should enable the loading of audit data to a database-based Audit Data Store.

As a part of the overall upgrade process, you should have created the IAU schema in the database where your other Oracle Fusion Middleware schemas reside. For more information about using the Audit Data Store, see [Managing the Audit Store](#) in *Securing Applications with Oracle Platform Security Services*.

Maintaining the Security Status of Older Java EE Web Service Applications

The introduction of global policy attachment support for Java EE web services and clients in 12c may impact the backwards compatibility of existing Java EE web services and clients (12.1.2 and earlier). If a Java EE web service or client endpoint that depends on the absence of a policy falls within the scope of a global policy attachment, the presence of the globally-attached policy can alter the security behavior of that endpoint.

Note:

In Fusion Middleware 12.1.2 and earlier, global policy attachments defined for SOAP Web Service and SOAP Web Service Client subject types were applicable to Oracle Infrastructure web services and clients only, and were ignored by Java EE web services and clients. After upgrading to 12c (12.2.1), the global policy attachments defined for these subject types apply to Java EE web services and clients, as well, and may alter the security behavior of existing Java EE web services and clients.

To maintain backwards compatibility, you can disable the global policy attachments for specific endpoints by attaching an OWSM no behavior policy to the service or client, such as `no_authentication_service_policy`, `no_authorization_service_policy`, or `no_messageprotection_service_policy`. See *Disabling a Globally Attached Policy in Securing Web Services and Managing Policies with Oracle Web Services Manager*.

Note:

You can use the WebLogic `Wssp1.5-No-Op.xml` no behavior policy. However, since WebLogic security policies can only be attached to web service clients programmatically, it requires code change. See *Disabling a Globally Attached Policy in Securing WebLogic Web Services for Oracle WebLogic Server*.

Documentation Resources for Managing your Oracle Fusion Middleware 12c Software

This topic provides a list of common administration tasks you likely want to perform after upgrading to Infrastructure 12c and associated documentation resources.

[Table 5-1](#) lists some common administration tasks you will likely want to perform after upgrading to Infrastructure 12c.

Table 3-11 Basic Administration Tasks

Task	Description	More Information
Getting familiar with Fusion Middleware administration tools	Get familiar with the various tools available which you can use to manage your environment.	Overview of Oracle Fusion Middleware Administration Tools in <i>Administering Oracle Fusion Middleware</i> .
Starting and stopping products and servers	Learn how to start and stop Oracle Fusion Middleware, including the Administration Server, Managed Servers, and components.	Starting and Stopping Oracle Fusion Middleware in <i>Administering Oracle Fusion Middleware</i> .
Configuring Secure Sockets Layer (SSL)	Learn how to set up secure communications among between Oracle Fusion Middleware components using SSL.	Configuring SSL in Oracle Fusion Middleware in <i>Administering Oracle Fusion Middleware</i> .
Monitoring Oracle Fusion Middleware	Learn how to keep track of the status of Oracle Fusion Middleware components.	Monitoring Oracle Fusion Middleware in <i>Administering Oracle Fusion Middleware</i> .
Understanding Backup and Recovery Procedures	Learn the recommended backup and recovery procedures for Oracle Fusion Middleware.	Introducing Backup and Recovery in <i>Administering Oracle Fusion Middleware</i> .

Using Your 11g Application Deployments in Oracle Fusion Middleware 12c

After you upgrade to Oracle Fusion Middleware 12c, the custom Java and Application Development Framework (ADF) you previously deployed on Oracle Fusion Middleware 11g work as they did in Oracle Fusion Middleware 11g. However, there are some new features and capabilities available in ADF 12c and in JDeveloper 12c.

The following sections provide some additional information about how you can migrate your applications to JDeveloper 12c:

- [About Oracle Application Development Framework \(ADF\) 12c](#)
Oracle ADF is an end-to-end Java EE framework that simplifies application development by providing out-of-the-box infrastructure services and a visual and declarative development experience.
- [About Oracle JDeveloper 12c](#)
Oracle JDeveloper is an integrated development environment that simplifies the development of Java-based applications addressing every step of the application lifecycle. JDeveloper offers complete end-to-end development for Oracle's platform and applications.

About Oracle Application Development Framework (ADF) 12c

Oracle ADF is an end-to-end Java EE framework that simplifies application development by providing out-of-the-box infrastructure services and a visual and declarative development experience.

Information about Oracle ADF can be found in the following Oracle Fusion Middleware 12c documentation library:

- Overview of Oracle ADF in *Understanding Oracle Application Development Framework*
- Oracle Application Development Framework (ADF) Common tasks page

About Oracle JDeveloper 12c

Oracle JDeveloper is an integrated development environment that simplifies the development of Java-based applications addressing every step of the application lifecycle. JDeveloper offers complete end-to-end development for Oracle's platform and applications.

- [Installing Oracle JDeveloper 12c](#)
Oracle JDeveloper provides an embedded version of Oracle WebLogic Server that can be used to locally test your applications.
- [Migrating Applications Using Oracle JDeveloper 12c](#)
After you install Oracle JDeveloper 12c, you can open your custom application projects in Oracle JDeveloper 12c and automatically migrate them to Oracle JDeveloper 12c.
- [About Migrating Asynchronous Web Services with Oracle JDeveloper 12c](#)
If your application contains Application Development Framework Business Components (ADF BC) asynchronous Web Services, ensure that you rebuild it using Oracle JDeveloper or the `ojdeploy` command line tool to generate the required deployment descriptors in your deployment archive.

Installing Oracle JDeveloper 12c

Oracle JDeveloper provides an embedded version of Oracle WebLogic Server that can be used to locally test your applications.

To install Oracle JDeveloper 12c, see *Installing the Oracle JDeveloper Software in Installing Oracle JDeveloper*.

For more information about using JDeveloper to test your applications, see *Deploying and Testing Applications Developed in Oracle JDeveloper in Installing Oracle JDeveloper*.

Migrating Applications Using Oracle JDeveloper 12c

After you install Oracle JDeveloper 12c, you can open your custom application projects in Oracle JDeveloper 12c and automatically migrate them to Oracle JDeveloper 12c.

See *Migrating Oracle JDeveloper From a Previous Version in Installing Oracle JDeveloper*.

About Migrating Asynchronous Web Services with Oracle JDeveloper 12c

If your application contains Application Development Framework Business Components (ADF BC) asynchronous Web Services, ensure that you rebuild it using Oracle JDeveloper or the `ojdeploy` command line tool to generate the required deployment descriptors in your deployment archive.

For more information about developing asynchronous Web Services, see Introduction to Developing Asynchronous Web Services in *Developing Oracle Infrastructure Web Services*.

If Your Existing Environment is a Clustered Configuration...

If your existing environment is a clustered configuration, then you must apply the changes to other cluster members in the domain by using the pack and unpack utilities.

Packing the Domain on the Primary Node

To pack the domain on the primary node:

1. Sign in to the primary node.
2. Pack the domain as shown in the following example:

```
./pack.sh -managed=true -domain=$DOMAIN_HOME/user_projects/  
domains/base_domain -template=sampledomaintemplate.jar -  
template_name=sample_domain_template
```

Unpacking the Domain on the Secondary Node

To unpack the domain on the secondary node:

1. Sign in to the secondary node.
2. Unpack the *sampledomaintemplate.jar* file containing the domain as shown in the following example:

```
./unpack.sh -domain=$DOMAIN_HOME/user_projects/domains/  
base_domain -template=$ORACLE_HOME/oracle_common/common/bin/  
smpdomaintemplate.jar -app_dir=$DOMAIN_HOME/user_projects/  
applications -overwrite_domain=true
```

Starting Servers and Processes

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

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

Note:

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

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

Step 1: Start the Administration Server

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

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

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

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

Step 2: Start Node Manager

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

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

Step 3: Start Oracle Identity Management Components

Start any Oracle Identity Management components, such as Oracle Internet Directory, that form part of your environment:

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

Step 4: Start the Managed Servers

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

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

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.

Step 5: Start System Components

To start system components, such as Oracle HTTP Server, use the `startComponent` script:

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

You can start system components in any order.

4

Upgrading Oracle Fusion Middleware Infrastructure from a Previous 12c Release

You can upgrade to Oracle Fusion Middleware Infrastructure 12c (12.2.1.3.0) from a previous 12c release.

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

- [About the Oracle Fusion Middleware Infrastructure Upgrade Process \(from a Previous 12c Release\)](#)
Review the flowchart and roadmap for an overview of the upgrade process for Oracle Fusion Middleware Infrastructure from a previous 12c release.
- [Installing Oracle Fusion Middleware Infrastructure](#)
Installing Fusion Middleware Infrastructure creates an Oracle home directory and lays down supporting software to install other Fusion Middleware products.
- [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.
- [Stopping Servers and Processes](#)
Before running the Upgrade Assistant, shut down all Oracle Fusion Middleware Managed Servers, Administration Servers, and system components (such as OHS) that may be using the schemas or configurations you want to update. Failure to do so may result in an incomplete or failed upgrade.
- [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.
- [Starting Servers and Processes](#)
After a successful upgrade, restart all processes and servers, including the Administration Server and any Managed Servers.
- [Using the Upgrade Validation Checklist](#)
After the upgrade, make sure that you can successfully complete the basic administration tasks, such as verifying whether you are able to start the Node Manager, Administration Server, Webtier, Administration Console, and Enterprise Manager Fusion Middleware Control.

- [Reapplying Custom Configuration Settings to setDomainEnv](#)
To complete the upgrade of your application environment to 12c it might be necessary to reapply any custom configuration settings to startup scripts, such as `setDomainEnv`. During the upgrade, the scripts are overwritten with new 12c versions. You must manually reapply the custom configuration settings you had made in previous releases.
- [Maintaining the Security Status of Older Java EE Web Service Applications](#)
The introduction of global policy attachment support for Java EE web services and clients in 12c may impact the backwards compatibility of existing Java EE web services and clients (12.1.2 and earlier). If a Java EE web service or client endpoint that depends on the absence of a policy falls within the scope of a global policy attachment, the presence of the globally-attached policy can alter the security behavior of that endpoint.
- [Documentation Resources for Managing your Oracle Fusion Middleware 12c Software](#)
This topic provides a list of common administration tasks you likely want to perform after upgrading to Infrastructure 12c and associated documentation resources.
- [If Your Existing Environment is a Clustered Configuration...](#)
If your existing environment is a clustered configuration, then you must apply the changes to other cluster members in the domain by using the `pack` and `unpack` utilities.
- [Starting Servers and Processes](#)
After a successful upgrade, restart all processes and servers, including the Administration Server and any Managed Servers.

About the Oracle Fusion Middleware Infrastructure Upgrade Process (from a Previous 12c Release)

Review the flowchart and roadmap for an overview of the upgrade process for Oracle Fusion Middleware Infrastructure from a previous 12c release.

Figure 4-1 Upgrade Process Flowchart for Oracle Fusion Middleware Infrastructure from a Previous 12c Release

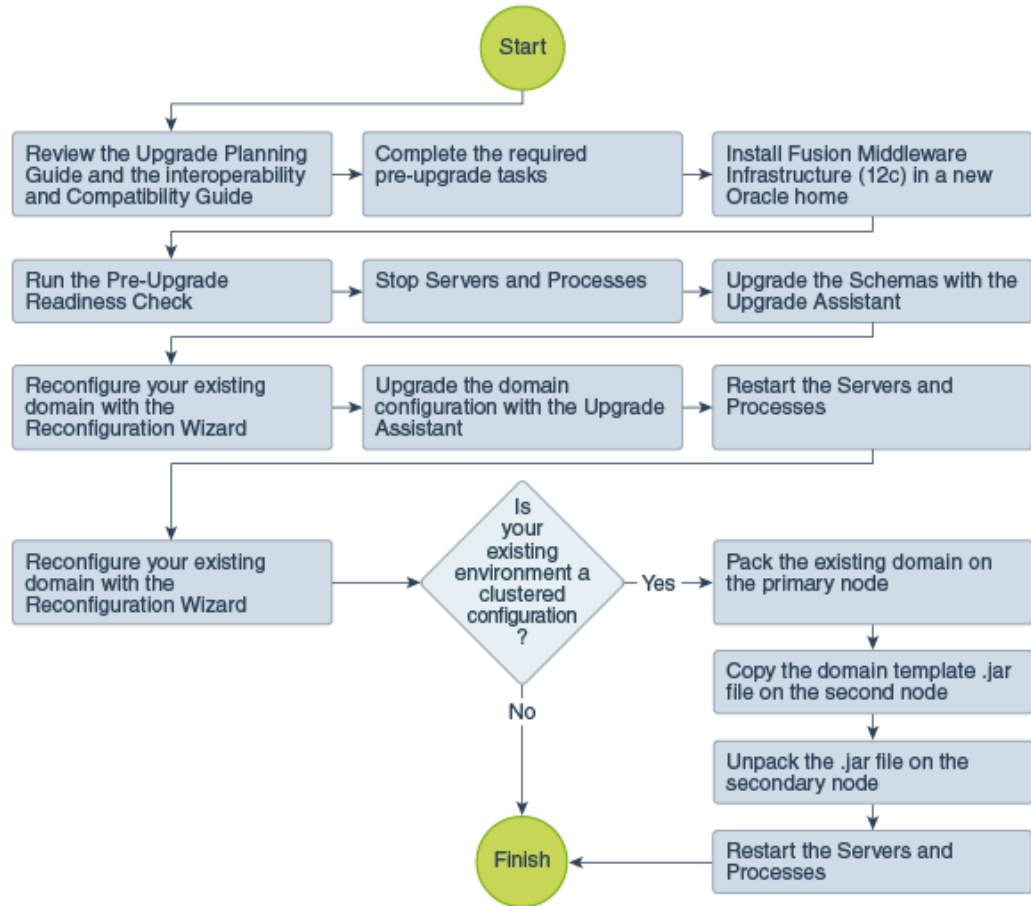


Table 4-1 lists the high-level steps that you need to perform to upgrade to Oracle Fusion Middleware Infrastructure Release 12.2.1.3.0:

Table 4-1 Tasks for Upgrading Oracle Fusion Middleware Infrastructure from a Previous 12c Release

Task	Description
<p>Optional</p> <p>Learn about the interoperability and compatibility factors that could affect how you upgrade to Oracle Fusion Middleware Infrastructure 12.2.1.3.0.</p>	<p>It is important to understand how two or more Oracle Fusion Middleware products of the same version or different versions work together (interoperate) in a supported Oracle Fusion Middleware configuration. You can learn more about interoperability and compatibility in Oracle® Fusion Middleware Understanding Interoperability and Compatibility.</p>
<p>Required</p> <p>If you have not done so already, review the introductory topics in this guide and complete the required pre-upgrade tasks.</p>	<p>The pre-upgrade tasks include cloning your production environment, verifying system requirements and certifications, purging unused data, and creating non-SYSDBA user. For a complete list of pre-upgrade tasks, see Pre-Upgrade Checklist.</p>

Table 4-1 (Cont.) Tasks for Upgrading Oracle Fusion Middleware Infrastructure from a Previous 12c Release

Task	Description
<p>Required Download and install the 12.2.1.3.0 Fusion Middleware Infrastructure distribution.</p>	<p>The Infrastructure distribution packs the WebLogic Server and the Java Required Files (JRF) that are required to set up the foundation to install other Fusion Middleware products.</p> <p>As per the upgrade topology defined in this guide, you must install the Infrastructure in a new Oracle home. Therefore, follow the procedure described in Installing Oracle Fusion Middleware Infrastructure.</p>
<p>Optional Run the Readiness Check.</p>	<p>Running the Readiness Check helps using the Upgrade Assistant helps you to determine whether your pre-upgrade environment is ready for upgrade. For the complete procedure, see Running a Pre-Upgrade Readiness Check.</p>
<p>Required Stop the servers and processes.</p>	<p>Before starting the upgrade process, stop all the servers, components, and processes. See Stopping Servers and Processes .</p>
<p>Required Upgrade the schemas with the Upgrade Assistant.</p>	<p>The Upgrade Assistant allows you to select All schemas used by a domain option. When you select this option, the Upgrade Assistant automatically selects all the schemas that are available for upgrade within that domain. See Upgrading Product Schemas.</p>
<p>Required Reconfigure the existing domain with the Reconfiguration Wizard.</p>	<p>When you run the Reconfiguration Wizard on your existing domain, it prepares your domain for upgrade by selecting and applying the reconfiguration templates. It also tests the JDBC data sources and component schemas that are present within your domain. To reconfigure you domain, follow the procedure described in About Reconfiguring the Domain.</p>
<p>Required Upgrade the existing domain configurations with the Upgrade Assistant.</p>	<p>After you have reconfigured your existing domain, you must run the Upgrade Assistant to upgrade all configurations used by your existing domain. You can see all the components within your domain that will be upgraded on the Component List screen when you run the Upgrade Assistant. See Upgrading Domain Component Configurations.</p>
<p>Required Restart the servers and processes.</p>	<p>The upgrade process is complete. You can now restart the servers, components, and processes. See Starting Servers and Processes.</p>
<p>Required Perform the post-upgrade tasks.</p>	<p>For a list of post-upgrade tasks, see Using the Upgrade Validation Checklist.</p>
<p>Required only if your existing environment is a clustered configuration Pack the existing domain on the primary node.</p>	<p>Run the <code>pack.sh cmd</code> script to pack your existing domain in a <code>domaintemplate.jar</code> file on the primary node. See If Your Existing Environment is a Clustered Configuration...</p>

Table 4-1 (Cont.) Tasks for Upgrading Oracle Fusion Middleware Infrastructure from a Previous 12c Release

Task	Description
Required only if your existing environment is a clustered configuration Copy the domain template.jar file on the secondary node.	Copy the <i>domaintemplate.jar</i> file on the secondary node so that you can unpack the contents of the file on the secondary node. See If Your Existing Environment is a Clustered Configuration...
Required only if your existing environment is a clustered configuration Unpack the jar file on the secondary node	Run the <code>unpack.sh cmd</code> script to unpack the <i>domaintemplate.jar</i> file on the secondary node. See If Your Existing Environment is a Clustered Configuration...
Required only if your existing environment is a clustered configuration Restart the servers and processes.	The upgrade process is complete. You can now restart the servers, components, and processes. See Starting Servers and Processes .

Installing Oracle Fusion Middleware Infrastructure

Installing Fusion Middleware Infrastructure creates an Oracle home directory and lays down supporting software to install other Fusion Middleware products.

The supported JDK version for 12c (12.2.1.3.0) is 1.8.0_131. Make sure that you upgrade to the latest JDK version before installing the 12c (12.2.1.3.0) software.

Note:

If you are upgrading from a previous 12c release (12.1.3.0, 12.2.1.0 or 12.2.1.1), then you must install the 12c (12.2.1.3.0) distributions into a new Oracle home. Do not attempt to reuse the existing Oracle home for this upgrade. Upgrading to 12c (12.2.1.3.0) is not a patch release.

To install Oracle Fusion Middleware Infrastructure distribution:

1. Sign in to the target system.
2. Download the Oracle Fusion Middleware Infrastructure (`fmw_12.2.1.3.0_infrastructure_generic.jar`) from [Oracle Technology Network](#) or [Oracle Software Delivery Cloud](#) to your target system.
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) `JDK_HOME/bin/java -jar fmw_12.2.1.3.0_infrastructure_generic.jar`
 - (Windows) `JDK_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 *Understanding Directories for Installation and Configuration in Oracle Fusion Middleware Planning an Installation of Oracle Fusion Middleware*.

9. On the Installation Type screen, select **Fusion Middleware Infrastructure** and 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 Security Updates screen, enter your My Oracle Support account information so you can receive the latest product information and security updates via your My Oracle Support account.

This screen appears the first time you install an Oracle product on a host.

If you do not have an Oracle Support account and you are sure that you want to skip this step, clear the check box and verify your selection in the follow-up dialog box.

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

If you want to save these options to a response file, click **Save Response File** and enter the response file location and name. The response file collects and stores all the information that you have entered, and enables you to perform a silent installation (from the command line) at a later time.

Click **Install** to begin the installation.

13. On the Installation Progress screen, when the progress bar displays 100%, click **Finish** to dismiss the installer, or click **Next** to see a summary.
14. The Installation Complete screen displays the Installation Location and the Feature Sets that are installed. Review this information and click **Finish** to close the installer.

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) `NEW_ORACLE_HOME/oracle_common/upgrade/bin`
 - (Windows) `NEW_ORACLE_HOME\oracle_common\upgrade\bin`
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, set the `DISPLAY` environment variable to the system name or IP address of your local workstation, and rerun Upgrade Assistant.

If you continue to receive these errors after setting `DISPLAY`, try launching another GUI tool, such as `vncconfig`. If you see the same errors, your `DISPLAY` environment variable may still not be set correctly.

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

Table 4-2 (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>NEW_ORACLE_HOME/ oracle_common/upgrade/ logs NEW_ORACLE_HOME/ oracle_common/upgrade/ temp</pre> <p>(Windows)</p> <pre>NEW_ORACLE_HOME\oracle_c ommon\upgrade\logs NEW_ORACLE_HOME\oracle_c ommon\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 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 March 30 11:15:52 EDT
2019
```

```
Log file is located at: ORACLE_HOME/oracle_common/upgrade/logs/
ua2016-05-30-11-14-06AM.log
```

```
Readiness Check Report File: 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 12.2.1.4.0.
```

```
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 12c Enterprise
Edition Release 12.2.1.4.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing
options
  Completed schema test: TEST_DATABASE_VERSION --> Test that the
database server version number is supported for upgrade +++ PASS
  Finished readiness check of Oracle Metadata Services with status:
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:

```

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_INDEXES --> 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 (12.2.1.3.0) by using the RCU.

Stopping Servers and Processes

Before running the Upgrade Assistant, shut down all Oracle Fusion Middleware Managed Servers, Administration Servers, and system components (such as OHS) that may be using the schemas or configurations you want to update. Failure to do so may result in an incomplete or failed upgrade.

If you are running the Node Manager, you should also stop the Node Manager. You can do this by closing the console window in which the Node Manager is running, or by using the `stopNodeManager WLST` command.

For instructions to stop an Oracle Fusion Middleware environment, see *Stopping an Oracle Fusion Middleware Environment* in *Administering 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.

- [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 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.

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.

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

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

- [Upgrade Assistant Parameters](#)

Upgrade Assistant Parameters

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

Table 4-4 Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
-readiness	Required for readiness checks Note: Readiness checks cannot be performed on standalone installations (those not managed by the WebLogic Server).	Performs the upgrade readiness check without performing an actual upgrade. Schemas and configurations are checked. Do not use this parameter if you have specified the -examine parameter.
-threads	Optional	Identifies the number of threads available for concurrent schema upgrades or readiness checks of the schemas. The value must be a positive integer in the range 1 to 8. The default is 4.
-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"> • TRACE • NOTIFICATION • WARNING • ERROR • INCIDENT_ERROR The default logging level is NOTIFICATION. Consider setting the -logLevel TRACE attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if -logLevel TRACE is used.

Table 4-4 (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>NEW_ORACLE_HOME/ oracle_common/upgrade/ logs NEW_ORACLE_HOME/ oracle_common/upgrade/ temp</pre> <p>(Windows)</p> <pre>NEW_ORACLE_HOME\oracle_c ommon\upgrade\logs NEW_ORACLE_HOME\oracle_c ommon\upgrade\temp</pre>
<code>-help</code>	Optional	Displays all of the command-line options.

Upgrading Schemas Using the Upgrade Assistant

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

To upgrade product schemas with the Upgrade Assistant:

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

Note:

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

2. On the Selected Schemas screen, select **All Schemas Used by a Domain**.
 - **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.

Click **Next**.

3. If you selected **All Schemas Used by a Domain**: On the Component List screen you will see two lists of schemas. The first list shows the components whose schemas are present in the domain and will be upgraded. The second list shows the list of missing schemas that may need to be created. If none of the required schemas are missing, you will only see the first list. Review both lists and click **Next**.

The Upgrade Assistant will attempt to create any missing schemas using the schema credentials used to create the existing domain schemas. You do not to launch the Repository Creation Utility.

If you want to exclude some components or schemas from the list, navigate back to the All Schemas screen and select **Individually Selected Schemas**. This option allows you to select only those schemas you want included in the upgrade.

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 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.

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 schemas that will be upgraded and/or created.

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 **Next** .

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 `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 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. Some schemas do not require an upgrade to this release and will retain their pre-upgrade version number.

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

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

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 Management (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/init-info/domain-info.xml` that are similar to this example:

```
<!--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/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 Domain with the Reconfiguration Wizard](#)
The Reconfiguration Wizard reconfigures the domain while retaining the location of the 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 domain directory:

1. Copy the source domain to a separate location to preserve the contents.
(Windows) `copy C:\domains\mydomain to C:\domains\mydomain_backup.`
(UNIX) `cp mydomain /domains/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 collocated managed servers before starting the reconfiguration process. See [Stopping Servers and Processes](#) .

To start the Reconfiguration Wizard in graphical mode:

1. Sign in to the system on which the domain resides.
2. Open the command shell (on UNIX operating systems) or open a command prompt window (on Windows operating systems).
3. **Edition Based Database Users Only:** If your schemas are configured with EBR database, a default edition name must be manually supplied before you run the Reconfiguration Wizard.

Run the following SQL command to set the default edition:

```
ALTER DATABASE DEFAULT EDITION = edition_name;
```

where *edition_name* is the child edition name.

4. Go to the `oracle_common/common/bin` directory:
 - (UNIX) `NEW_ORACLE_HOME/oracle_common/common/bin`
 - (Windows) `NEW_ORACLE_HOME\oracle_common\commom\bin`
5. 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.



Note:

When you run this command, the following error message might appear to indicate that the default cache directory is not valid:

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

You can change the cache directory by setting the environment variable `CONFIG_JVM_ARGS`. For example:

```
CONFIG_JVM_ARGS=-Dpython.cachedir=valid_directory
```

Reconfiguring the Domain with the Reconfiguration Wizard

The Reconfiguration Wizard reconfigures the domain while retaining the location of the domain. Navigate through the screens in the Reconfiguration Wizard to reconfigure your existing domain.

! Important:

If the source is a clustered environment, run the Reconfiguration Wizard on the primary node only. Use the pack/unpack utility to apply the changes to other cluster members in the domain as described in [If Your Existing Environment is a Clustered Configuration...](#)

To reconfigure the domain:

1. Sign in to the system on which the domain resides.
2. **Edition Based Database Users Only:** If you have configured your schemas with Edition-Based Reassociation, you must manually supply a default edition name before running the Reconfiguration Wizard.

To set the default edition, enter the following SQL command:

```
ALTER DATABASE DEFAULT EDITION = edition_name;
```

Where, *edition_name* is the name of the default database edition.

3. 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**.
4. 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.
5. 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.

6. On the JDBC Data Sources screen, configure the JDBC data sources defined in your domain source.

The JDBC data sources associated with the products for which you are creating the domain are listed in the lower half of the screen. A JDBC data source contains a pool of database connections that are created when the data source instance is created, deployed or targeted, or at server startup. Applications look up a data source on the JNDI tree, and then request a connection. When the applications no longer need the connections, they return the connections to the connection pool in the data source.

From the **Data Source Name** drop-down list, select the data source(s) for which you want to specify the settings. The values that you specify are displayed in the appropriate columns in the data source list, for the selected data source.

For Oracle RAC Configuration for data sources, you can select one of the three options:

- Convert to GridLink
- Convert to RAC multi data source
- Don't convert

For more information about each option, click **Help**.

After specifying the details, click **Next**.

If you do not select any data sources on the JDBC Data Sources screen, the following warning displays:

Missing Driver

Click **Ok** to proceed without verification, click **Cancel** to return to the JDBC Data Sources page.

In this case, if you click **Ok**, the data sources are not verified.

7. On the JDBC Data Sources Test screen, select the check box for the data source connection you configured on the JDBC Data Sources screen and click **Test Selected Connections** to test the data source connection.

 **Note:**

To test the database connections, the database to which you are connecting must be running. If you do not want to test the connections at this time, do not select any data sources. Click **Next** to continue.

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

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

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

 **Note:**

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

 **Note:**

For any existing 12c (12.2.1.2.0) datasource, the reconfiguration will preserve the existing values. For new datasources where the schema was created for 12c (12.2.1.3.0) by the RCU, the default connection data will be retrieved from the `_STB` schema. If no connection data for a given schema is found in the `_STB` schema, then the default connection data is used.

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

 **Note:**

If you are upgrading from 12c (12.2.1.2.0), and your database has `_OPSS` or `_IAU` 12c (12.2.1.2.0) database schemas, you must manually enter database connection details for those schemas. These schemas were not required in 12c and had to be created manually. Users could assign any name to these schemas, therefore the Reconfiguration Wizard does not recognize them. When providing connection information for `_IAU`, use the `IAU_APPEND` user information.

9. On the JDBC Component Schema Test screen, select all the component schemas and click **Test Selected Connections** to test the connection for each schema. The result of the test is indicated in the Status column.

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

10. The Node Manager screen is only displayed if the domain you are reconfiguring is currently using a per host Node Manager.

On the Node Manager screen, select the Node Manager configuration to use for the reconfigured domain. The resulting configuration depends on the combination of options you select for **Node Manager Type** and **Node Manager Configuration**.

Table 4-5 Field Descriptions for Node Manager Screen

Option	Description
Per Domain Default Location	If you select this option, the Node Manager home is redefined to <code>\$domain_name/nodemanager</code> and you cannot edit the Node Manager home.

Table 4-5 (Cont.) Field Descriptions for Node Manager Screen

Option	Description
Per Domain Custom Location	Select this option if you want the per domain Node Manager configuration files to be created in a specific location for this domain. Specify the directory in the Node Manager Home field, or click Browse to use the navigation tree to select the location. The specified directory must be empty. The <code>nodemanager.properties</code> and <code>nodemanager.domains</code> files are created in this directory.
Node Manager Home	If you selected the Per Domain Custom Location option, click Browse to navigate to the directory location that you want to use to store the per domain Node Manager configuration.
Manual Node Manager Setup	If you select this option, creation of the Node Manager configuration for the domain is skipped (all remaining fields cannot be modified), and if you want to use Node Manager in the domain, you must manually configure Node Manager as described in <i>Completing the Node Manager Configuration</i> . The reconfigured domain will still use a per host Node Manager configuration. You should also select this option if your existing domain is not configured to use Node Manager and you do not want to use Node Manager in the reconfigured domain. For more information about Node Manager configuration, see <i>Administering Node Manager for Oracle WebLogic Server</i> .
Node Manager Configuration	Select one of the following two options. These fields are not available if you selected Manual Node Manager Setup .
Create New Configuration	A per domain Node Manager configuration will be automatically created for the reconfigured domain using default settings in <code>nodemanager.properties</code> . If necessary, you can modify <code>nodemanager.properties</code> after the domain has been successfully reconfigured.
Migrate Existing Configuration	The existing per host Node Manager configuration will be migrated to a per domain configuration for the reconfigured domain. This does not include environment-specific settings for <code>ListenAddress</code> , <code>ListenPort</code> , <code>StartScriptName</code> , <code>JavaHome</code> , and <code>LogFile</code> .
Node Manager Home	If you selected the Migrate Existing Configuration option, enter or browse to the Node Manager home directory that you want to migrate to the reconfigured domain.

Table 4-5 (Cont.) Field Descriptions for Node Manager Screen

Option	Description
Apply Oracle Recommended Defaults	<p>If you selected the Migrate Existing Configuration option, select this check box if you want to use Oracle-recommended defaults in the <code>nodemanager.properties</code> file. Deselect this check box if you want to continue using the settings in the <code>nodemanager.properties</code> file being migrated.</p> <p>Oracle-recommended properties with default values are as follows:</p> <pre> LogLimit=0 AuthenticationEnabled=true LogLevel=INFO DomainsFileEnabled=true NativeVersionEnabled=true LogToStderr=true SecureListener=true LogCount=1 StopScriptEnabled=false QuitEnabled=false LogAppend=true StateCheckInterval=500 CrashRecoveryEnabled=false StartScriptEnabled=true LogFormatter=weblogic.nodemanager.server.LogFormatter ListenBacklog=50 </pre>
Node Manager Credentials: Username, Password	Specify the username and password that you want to use to start Node Manager in the reconfigured domain.

- 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**.

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

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

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

 **Note:**

The location of the domain does not change when you reconfigure it.

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

During this process:

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

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

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

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

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

For all the Advanced Configurations-related screens, such as Managed Servers, Clusters, Machines, HTTP Proxy Applications, Coherence Clusters, and System Components, see *Upgrading Oracle WebLogic Server*.

If an error occurs while reconfiguring your domain, refer to the Important Notes About the Domain Upgrade Process in *Upgrading Oracle WebLogic Server*.

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 the Domain Configurations with the Upgrade Assistant](#)
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.

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

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

- [Upgrade Assistant Parameters](#)

Upgrade Assistant Parameters

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

Table 4-6 Upgrade Assistant Command-Line Parameters

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

Table 4-6 (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">• TRACE• NOTIFICATION• WARNING• ERROR• INCIDENT_ERROR The default logging level is NOTIFICATION. Consider setting the -logLevel TRACE attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if -logLevel TRACE is used.

Table 4-6 (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>NEW_ORACLE_HOME/ oracle_common/upgrade/ logs NEW_ORACLE_HOME/ oracle_common/upgrade/ temp</pre> <p>(Windows)</p> <pre>NEW_ORACLE_HOME\oracle_c ommon\upgrade\logs NEW_ORACLE_HOME\oracle_c ommon\upgrade\temp</pre>
<code>-help</code>	Optional	Displays all of the command-line options.

Upgrading the Domain Configurations with the Upgrade Assistant

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 All Configurations screen, select **All Configurations Used by a Domain** and specify your domain location in the **Domain Directory** field by entering it directly or by clicking **Browse** to use a navigation tree to select a valid 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 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.

7. 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.

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 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**.

9. 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 Oracle Enterprise Manager Fusion Middleware Control Console, go to:
`http://administration_server_host:administration_server_port/em`

 **Note:**

After upgrade, make sure you run the administration tools from the new 12c (12.2.1.3.0) Oracle home directory and not from the previous Oracle home directory.

During the upgrade process, some OWSM documents, including policy sets and predefined documents such as policies and assertion templates, may need to be upgraded. If a policy set or a predefined document is upgraded, its version number is incremented by 1.

If you created the FMW user to run the Upgrade Assistant, ensure that you delete the account after verifying your upgrade was successful.

Starting Servers and Processes

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

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

 **Note:**

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

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

Step 1: Start the Administration Server

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

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

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

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

Step 2: Start Node Manager

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

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

Step 3: Start Oracle Identity Management Components

Start any Oracle Identity Management components, such as Oracle Internet Directory, that form part of your environment:

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

Step 4: Start the Managed Servers

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

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

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.

Step 5: Start System Components

To start system components, such as Oracle HTTP Server, use the `startComponent` script:

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

You can start system components in any order.

Using the Upgrade Validation Checklist

After the upgrade, make sure that you can successfully complete the basic administration tasks, such as verifying whether you are able to start the Node Manager, Administration Server, Webtier, Administration Console, and Enterprise Manager Fusion Middleware Control.

Note:

The order in which you start the following servers is important and failure to start (or stop) them in the correct order can cause issues with the deployment.

1. Verify that you are able to start the Node Managers.
2. Verify that you are able to start the Administration Server and any Managed Servers (if included) from the original Domain Home `bin` directory. Windows operating system users may find it useful to start the servers from a new command prompt (and not the one used to launch the 12c Upgrade Assistant).

 **Note:**

OHS does not need a Managed Server for its own configuration.

3. Verify that you are able to start the Webtier (OHS server).
4. Verify that you are able to access the Administration console and Enterprise Manager using the following URLs:

Administration console: `http://
machinename.my_company_com:administration_port/console`

Enterprise Manager: `http://
machinename.my_company_com:administration_port/em`

Reapplying Custom Configuration Settings to setDomainEnv

To complete the upgrade of your application environment to 12c it might be necessary to reapply any custom configuration settings to startup scripts, such as `setDomainEnv`. During the upgrade, the scripts are overwritten with new 12c versions. You must manually reapply the custom configuration settings you had made in previous releases.

See [Re-apply Customizations to Startup Scripts](#).

 **Note:**

To prevent losing your custom configuration settings in a future upgrade, see [Maintaining Your Custom setDomainEnv Settings](#).

Maintaining the Security Status of Older Java EE Web Service Applications

The introduction of global policy attachment support for Java EE web services and clients in 12c may impact the backwards compatibility of existing Java EE web services and clients (12.1.2 and earlier). If a Java EE web service or client endpoint that depends on the absence of a policy falls within the scope of a global policy

attachment, the presence of the globally-attached policy can alter the security behavior of that endpoint.

 **Note:**

In Fusion Middleware 12.1.2 and earlier, global policy attachments defined for SOAP Web Service and SOAP Web Service Client subject types were applicable to Oracle Infrastructure web services and clients only, and were ignored by Java EE web services and clients. After upgrading to 12c (12.2.1), the global policy attachments defined for these subject types apply to Java EE web services and clients, as well, and may alter the security behavior of existing Java EE web services and clients.

To maintain backwards compatibility, you can disable the global policy attachments for specific endpoints by attaching an OWSM no behavior policy to the service or client, such as `no_authentication_service_policy`, `no_authorization_service_policy`, or `no_messageprotection_service_policy`. See *Disabling a Globally Attached Policy in Securing Web Services and Managing Policies with Oracle Web Services Manager*.

 **Note:**

You can use the WebLogic `Wssp1.5-No-Op.xml` no behavior policy. However, since WebLogic security policies can only be attached to web service clients programmatically, it requires code change. See *Disabling a Globally Attached Policy in Securing WebLogic Web Services for Oracle WebLogic Server*.

Documentation Resources for Managing your Oracle Fusion Middleware 12c Software

This topic provides a list of common administration tasks you likely want to perform after upgrading to Infrastructure 12c and associated documentation resources.

[Table 5-1](#) lists some common administration tasks you will likely want to perform after upgrading to Infrastructure 12c.

Table 4-7 Basic Administration Tasks

Task	Description	More Information
Getting familiar with Fusion Middleware administration tools	Get familiar with the various tools available which you can use to manage your environment.	Overview of Oracle Fusion Middleware Administration Tools in <i>Administering Oracle Fusion Middleware</i> .
Starting and stopping products and servers	Learn how to start and stop Oracle Fusion Middleware, including the Administration Server, Managed Servers, and components.	Starting and Stopping Oracle Fusion Middleware in <i>Administering Oracle Fusion Middleware</i> .

Table 4-7 (Cont.) Basic Administration Tasks

Task	Description	More Information
Configuring Secure Sockets Layer (SSL)	Learn how to set up secure communications among between Oracle Fusion Middleware components using SSL.	Configuring SSL in Oracle Fusion Middleware in <i>Administering Oracle Fusion Middleware</i> .
Monitoring Oracle Fusion Middleware	Learn how to keep track of the status of Oracle Fusion Middleware components.	Monitoring Oracle Fusion Middleware in <i>Administering Oracle Fusion Middleware</i> .
Understanding Backup and Recovery Procedures	Learn the recommended backup and recovery procedures for Oracle Fusion Middleware.	Introducing Backup and Recovery in <i>Administering Oracle Fusion Middleware</i> .

If Your Existing Environment is a Clustered Configuration...

If your existing environment is a clustered configuration, then you must apply the changes to other cluster members in the domain by using the pack and unpack utilities.

Packing the Domain on the Primary Node

To pack the domain on the primary node:

1. Sign in to the primary node.
2. Pack the domain as shown in the following example:

```
./pack.sh -managed=true -domain=$DOMAIN_HOME/user_projects/
domains/base_domain -template=sampledomaintemplate.jar -
template_name=sample_domain_template
```

Unpacking the Domain on the Secondary Node

To unpack the domain on the secondary node:

1. Sign in to the secondary node.
2. Unpack the *sampledomaintemplate.jar* file containing the domain as shown in the following example:

```
./unpack.sh -domain=$DOMAIN_HOME/user_projects/domains/
base_domain -template=$ORACLE_HOME/oracle_common/common/bin/
smpledomaintemplate.jar -app_dir=$DOMAIN_HOME/user_projects/
applications -overwrite_domain=true
```

Starting Servers and Processes

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

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

 **Note:**

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

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

Step 1: Start the Administration Server

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

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

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

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

Step 2: Start Node Manager

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

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

Step 3: Start Oracle Identity Management Components

Start any Oracle Identity Management components, such as Oracle Internet Directory, that form part of your environment:

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

Step 4: Start the Managed Servers

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

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

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.

Step 5: Start System Components

To start system components, such as Oracle HTTP Server, use the `startComponent` script:

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

You can start system components in any order.

5

Tasks to Perform After Upgrade

The topics in this section describe the tasks you might have to perform after upgrading to Oracle Fusion Middleware Infrastructure 12c . Perform only those steps that apply to your upgraded environment.

- [Using the Upgrade Validation Checklist](#)
After the upgrade, make sure that you can successfully complete the basic administration tasks, such as verifying whether you are able to start the Node Manager, Administration Server, Webtier, Administration Console, and Enterprise Manager Fusion Middleware Control.
- [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.
- [Reapplying Custom Configuration Settings to setDomainEnv](#)
To complete the upgrade of your application environment to 12c it might be necessary to reapply any custom configuration settings to startup scripts, such as `setDomainEnv`. During the upgrade, the scripts are overwritten with new 12c versions. You must manually reapply the custom configuration settings you had made in previous releases.
- [Configuring an Oracle Fusion Middleware 12c Audit Data Store](#)
If you were using a file-based audit store in your 11g deployment, then after the upgrade to Oracle Fusion Middleware 12c, you should enable the loading of audit data to a database-based Audit Data Store.
- [Maintaining the Security Status of Older Java EE Web Service Applications](#)
The introduction of global policy attachment support for Java EE web services and clients in 12c may impact the backwards compatibility of existing Java EE web services and clients (12.1.2 and earlier). If a Java EE web service or client endpoint that depends on the absence of a policy falls within the scope of a global policy attachment, the presence of the globally-attached policy can alter the security behavior of that endpoint.
- [Documentation Resources for Managing your Oracle Fusion Middleware 12c Software](#)
This topic provides a list of common administration tasks you likely want to perform after upgrading to Infrastructure 12c and associated documentation resources.
- [Using Your 11g Application Deployments in Oracle Fusion Middleware 12c](#)
After you upgrade to Oracle Fusion Middleware 12c, the custom Java and Application Development Framework (ADF) you previously deployed on Oracle Fusion Middleware 11g work as they did in Oracle Fusion Middleware 11g. However, there are some new features and capabilities available in ADF 12c and in JDeveloper 12c.

Using the Upgrade Validation Checklist

After the upgrade, make sure that you can successfully complete the basic administration tasks, such as verifying whether you are able to start the Node Manager, Administration Server, Webtier, Administration Console, and Enterprise Manager Fusion Middleware Control.

 **Note:**

The order in which you start the following servers is important and failure to start (or stop) them in the correct order can cause issues with the deployment.

1. Verify that you are able to start the Node Managers.
2. Verify that you are able to start the Administration Server and any Managed Servers (if included) from the original Domain Home `bin` directory. Windows operating system users may find it useful to start the servers from a new command prompt (and not the one used to launch the 12c Upgrade Assistant).

 **Note:**

OHS does not need a Managed Server for its own configuration.

3. Verify that you are able to start the Webtier (OHS server).
4. Verify that you are able to access the Administration console and Enterprise Manager using the following URLs:

Administration console: `http://
machinename.my_company_com:administration_port/console`

Enterprise Manager: `http://
machinename.my_company_com:administration_port/em`

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

 **Note:**

After upgrade, make sure you run the administration tools from the new 12c (12.2.1.3.0) Oracle home directory and not from the previous Oracle home directory.

During the upgrade process, some OWSM documents, including policy sets and predefined documents such as policies and assertion templates, may need to be upgraded. If a policy set or a predefined document is upgraded, its version number is incremented by 1.

If you created the FMW user to run the Upgrade Assistant, ensure that you delete the account after verifying your upgrade was successful.

Reapplying Custom Configuration Settings to setDomainEnv

To complete the upgrade of your application environment to 12c it might be necessary to reapply any custom configuration settings to startup scripts, such as `setDomainEnv`. During the upgrade, the scripts are overwritten with new 12c versions. You must manually reapply the custom configuration settings you had made in previous releases.

See [Re-apply Customizations to Startup Scripts](#).

 **Note:**

To prevent losing your custom configuration settings in a future upgrade, see [Maintaining Your Custom setDomainEnv Settings](#).

Configuring an Oracle Fusion Middleware 12c Audit Data Store

If you were using a file-based audit store in your 11g deployment, then after the upgrade to Oracle Fusion Middleware 12c, you should enable the loading of audit data to a database-based Audit Data Store.

As a part of the overall upgrade process, you should have created the IAU schema in the database where your other Oracle Fusion Middleware schemas reside. For more information about using the Audit Data Store, see [Managing the Audit Store](#) in *Securing Applications with Oracle Platform Security Services*.

Maintaining the Security Status of Older Java EE Web Service Applications

The introduction of global policy attachment support for Java EE web services and clients in 12c may impact the backwards compatibility of existing Java EE web services and clients (12.1.2 and earlier). If a Java EE web service or client endpoint

that depends on the absence of a policy falls within the scope of a global policy attachment, the presence of the globally-attached policy can alter the security behavior of that endpoint.

 **Note:**

In Fusion Middleware 12.1.2 and earlier, global policy attachments defined for SOAP Web Service and SOAP Web Service Client subject types were applicable to Oracle Infrastructure web services and clients only, and were ignored by Java EE web services and clients. After upgrading to 12c (12.2.1), the global policy attachments defined for these subject types apply to Java EE web services and clients, as well, and may alter the security behavior of existing Java EE web services and clients.

To maintain backwards compatibility, you can disable the global policy attachments for specific endpoints by attaching an OWSM no behavior policy to the service or client, such as `no_authentication_service_policy`, `no_authorization_service_policy`, or `no_messageprotection_service_policy`. See *Disabling a Globally Attached Policy in Securing Web Services and Managing Policies with Oracle Web Services Manager*.

 **Note:**

You can use the WebLogic `Wssp1.5-No-Op.xml` no behavior policy. However, since WebLogic security policies can only be attached to web service clients programmatically, it requires code change. See *Disabling a Globally Attached Policy in Securing WebLogic Web Services for Oracle WebLogic Server*.

Documentation Resources for Managing your Oracle Fusion Middleware 12c Software

This topic provides a list of common administration tasks you likely want to perform after upgrading to Infrastructure 12c and associated documentation resources.

[Table 5-1](#) lists some common administration tasks you will likely want to perform after upgrading to Infrastructure 12c.

Table 5-1 Basic Administration Tasks

Task	Description	More Information
Getting familiar with Fusion Middleware administration tools	Get familiar with the various tools available which you can use to manage your environment.	Overview of Oracle Fusion Middleware Administration Tools in <i>Administering Oracle Fusion Middleware</i> .

Table 5-1 (Cont.) Basic Administration Tasks

Task	Description	More Information
Starting and stopping products and servers	Learn how to start and stop Oracle Fusion Middleware, including the Administration Server, Managed Servers, and components.	Starting and Stopping Oracle Fusion Middleware in <i>Administering Oracle Fusion Middleware</i> .
Configuring Secure Sockets Layer (SSL)	Learn how to set up secure communications among between Oracle Fusion Middleware components using SSL.	Configuring SSL in Oracle Fusion Middleware in <i>Administering Oracle Fusion Middleware</i> .
Monitoring Oracle Fusion Middleware	Learn how to keep track of the status of Oracle Fusion Middleware components.	Monitoring Oracle Fusion Middleware in <i>Administering Oracle Fusion Middleware</i> .
Understanding Backup and Recovery Procedures	Learn the recommended backup and recovery procedures for Oracle Fusion Middleware.	Introducing Backup and Recovery in <i>Administering Oracle Fusion Middleware</i> .

Using Your 11g Application Deployments in Oracle Fusion Middleware 12c

After you upgrade to Oracle Fusion Middleware 12c, the custom Java and Application Development Framework (ADF) you previously deployed on Oracle Fusion Middleware 11g work as they did in Oracle Fusion Middleware 11g. However, there are some new features and capabilities available in ADF 12c and in JDeveloper 12c.

The following sections provide some additional information about how you can migrate your applications to JDeveloper 12c:

- [About Oracle Application Development Framework \(ADF\) 12c](#)
Oracle ADF is an end-to-end Java EE framework that simplifies application development by providing out-of-the-box infrastructure services and a visual and declarative development experience.
- [About Oracle JDeveloper 12c](#)
Oracle JDeveloper is an integrated development environment that simplifies the development of Java-based applications addressing every step of the application lifecycle. JDeveloper offers complete end-to-end development for Oracle's platform and applications.

About Oracle Application Development Framework (ADF) 12c

Oracle ADF is an end-to-end Java EE framework that simplifies application development by providing out-of-the-box infrastructure services and a visual and declarative development experience.

Information about Oracle ADF can be found in the following Oracle Fusion Middleware 12c documentation library:

- Overview of Oracle ADF in *Understanding Oracle Application Development Framework*
- Oracle Application Development Framework (ADF) Common tasks page

About Oracle JDeveloper 12c

Oracle JDeveloper is an integrated development environment that simplifies the development of Java-based applications addressing every step of the application lifecycle. JDeveloper offers complete end-to-end development for Oracle's platform and applications.

- [Installing Oracle JDeveloper 12c](#)
Oracle JDeveloper provides an embedded version of Oracle WebLogic Server that can be used to locally test your applications.
- [Migrating Applications Using Oracle JDeveloper 12c](#)
After you install Oracle JDeveloper 12c, you can open your custom application projects in Oracle JDeveloper 12c and automatically migrate them to Oracle JDeveloper 12c.
- [About Migrating Asynchronous Web Services with Oracle JDeveloper 12c](#)
If your application contains Application Development Framework Business Components (ADF BC) asynchronous Web Services, ensure that you rebuild it using Oracle JDeveloper or the `ojdeploy` command line tool to generate the required deployment descriptors in your deployment archive.

Installing Oracle JDeveloper 12c

Oracle JDeveloper provides an embedded version of Oracle WebLogic Server that can be used to locally test your applications.

To install Oracle JDeveloper 12c, see *Installing the Oracle JDeveloper Software in Installing Oracle JDeveloper*.

For more information about using JDeveloper to test your applications, see *Deploying and Testing Applications Developed in Oracle JDeveloper in Installing Oracle JDeveloper*.

Migrating Applications Using Oracle JDeveloper 12c

After you install Oracle JDeveloper 12c, you can open your custom application projects in Oracle JDeveloper 12c and automatically migrate them to Oracle JDeveloper 12c.

See *Migrating Oracle JDeveloper From a Previous Version in Installing Oracle JDeveloper*.

About Migrating Asynchronous Web Services with Oracle JDeveloper 12c

If your application contains Application Development Framework Business Components (ADF BC) asynchronous Web Services, ensure that you rebuild it using Oracle JDeveloper or the `ojdeploy` command line tool to generate the required deployment descriptors in your deployment archive.

For more information about developing asynchronous Web Services, see *Introduction to Developing Asynchronous Web Services in Developing Oracle Infrastructure Web Services*.

A

Troubleshooting the Infrastructure Upgrade

If the Infrastructure upgrade fails, troubleshoot the cause using the log file and guidelines in this topic.

▲ Caution:

As with most Fusion Middleware errors, errors that are detected in the Examine phase can be fixed and the Upgrade Assistant can continue to run. Errors that occur during the Upgrade phase, however, must be corrected using the restored backup files and the upgrade process must be started from the beginning. Do not attempt to rerun an upgrade that fails during the Upgrade phase. The environment should be considered unstable and will need to be restored to its pre-upgrade state.

See General Troubleshooting Guidelines in *Upgrading with the Upgrade Assistant*.

- [Authentication Failure — JSchException: Auth Fail](#)
When Running the Upgrade Assistant to upgrade Weblogic Component Configurations, if you provide incorrect login credentials for a UMS server, you an exception in the Upgrade Assistant log files as shown in this topic.
- [Error while Copying User Messaging Service \(UMS\) Configuration Files](#)
If the Upgrade Assistant fails to automatically copy the UMS configuration files, you must stop the upgrade and manually copy the configuration files before attempting to upgrade UMS. This process is required only if the Upgrade Assistant fails to automatically copy the configuration files or if you prefer to copy the configuration files manually.

Authentication Failure — JSchException: Auth Fail

When Running the Upgrade Assistant to upgrade Weblogic Component Configurations, if you provide incorrect login credentials for a UMS server, you an exception in the Upgrade Assistant log files as shown in this topic.

```
[upgrade.UCSUMS.UCSUMS_CONFIGURATION_PLUGIN] [tid: 110] [ecid: 88ab893d-a523-4a83-b5a6-f7b1cf8cb029-00000001,0] [[com.jcraft.jsch.JSchException: Auth fail
```

The resolution to this error depends on when the error occurred:

If this error occurred during the **Examine phase** (before Upgrade phase): Verify that the username and password you entered are valid for all managed servers and directories and that the username provided has privileges for ssh. Once you have corrected the error, retry the connection.

If this error occurred during the **Upgrade phase**, your upgrade operation did not succeed and you need to restore your files from backup and start the upgrade again. Make sure that you use the correct server login credentials when prompted.

▲ Caution:

Errors that occur during the Upgrade phase are non-reentrant, meaning you cannot simply correct the error and continue through the upgrade. Once you click Upgrade, if an error occurs then the environment must be restored from backup before you start the upgrade process again.

Error while Copying User Messaging Service (UMS) Configuration Files

If the Upgrade Assistant fails to automatically copy the UMS configuration files, you must stop the upgrade and manually copy the configuration files before attempting to upgrade UMS. This process is required only if the Upgrade Assistant fails to automatically copy the configuration files or if you prefer to copy the configuration files manually.

This section describes the location of the UMS configuration files that are copied from the remote managed server nodes to the Admin server while upgrading UMS from 11g to 12c. Note that the Upgrade Assistant can automatically copy the remote configuration files, if all necessary prerequisites are met and the required login information is provided. For more information about using Upgrade Assistant to copy configuration files, see Identifying Configurations that can be Upgraded with the Upgrade Assistant in *Upgrading with the Upgrade Assistant*.

However, if the Upgrade Assistant cannot locate your files, then you must copy the configuration files from the remote managed server to the same location on the Administration server running the upgrade. The configuration files that must be copied include the UMS server configuration files (appconfig.xml), driver configuration files (driverconfig.xml), and the user preferences files (businesssterms.xml). These files are located in the /applications folder for each managed server, as shown in [Table A-1](#).

After manually copying the configuration files from the managed server to the Administration server, you must start the Upgrade Assistant again.

Table A-1 Configuration File locations

Configuration file	Location
UMS Server (appconfig.xml)	<code>DOMAIN_HOME/config/fmwconfig/servers/ MANAGED_SERVER_NAME/applications/ usermessagingserver/configuration/ appconfig.xml</code>
Driver Configuration (driverconfig.xml)	<code>DOMAIN_HOME/config/fmwconfig/servers/ MANAGED_SERVER_NAME/applications/ usermessagingdriver-DRIVER_NAME/ configuration/driverconfig.xml</code>

Table A-1 (Cont.) Configuration File locations

Configuration file	Location
User Preferences (businessterms.xml)	<i>DOMAIN_HOME</i> /config/fmwconfig/servers/ <i>MANAGED_SERVER_NAME</i> /applications/ usermessagingserver/configuration/ businessterms.xml

 **Note:**

If there are multiple drivers deployed in a domain, then you must ensure that configuration files for all drivers are copied. This can be achieved by replacing the *DRIVER_NAME* with as many drivers deployed in that domain.

B

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.

- [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
```