

Oracle® Fusion Middleware

Upgrading to the Oracle Fusion Middleware Infrastructure
12c (12.2.1.1)

E71494-01

June 2016

Documentation for Oracle Fusion Middleware administrators
who want to upgrade to the Oracle Fusion Middleware
Infrastructure 12c.

Oracle Fusion Middleware Upgrading to the Oracle Fusion Middleware Infrastructure, 12c (12.2.1.1)

E71494-01

Copyright © 2015, 2016, Oracle and/or its affiliates. All rights reserved.

Primary Authors: Priyanka Chheda, Lisa Jamen

Contributors: Bill Norcott, Michael Rubino, Renga Rengarajan, Robert Donat, Rubis Chowallur, Sitaraman Swaminathan.

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 installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. 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 Xeon 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, Opteron, the AMD logo, and the AMD Opteron 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	vii
Audience	vii
Documentation Accessibility	vii
Related Documents.....	vii
Conventions.....	viii
What's New in This Guide.....	ix
New and Changed Features for 12c (12.2.1.1)	ix
Other Changes to This Guide.....	ix
1 Introduction to Oracle Fusion Middleware Infrastructure Upgrade	
1.1 Understanding the Starting Points for an Oracle Fusion Middleware Infrastructure Upgrade	1-1
1.1.1 About Oracle Fusion Middleware Infrastructure 12c	1-2
1.1.2 Key Differences Between Application Developer 11g and Infrastructure 12c.....	1-2
1.2 Upgrading Security Store	1-5
1.3 Understanding the Standard Upgrade Topologies for Infrastructure.....	1-6
1.3.1 Fusion Middleware Infrastructure Standard Upgrade Topology.....	1-6
1.4 Understanding the Additional New Features for Oracle Fusion Middleware 12c.....	1-9
1.5 Flow Chart and Task Roadmaps for Upgrading to Fusion Middleware Infrastructure.....	1-10
1.5.1 Flow Chart for Upgrading the Infrastructure Standard Upgrade Topologies.....	1-10
1.5.2 Roadmap for Upgrading the Infrastructure Standard Upgrade Topologies.....	1-11
1.6 Upgrading Custom Applications Using Oracle JDeveloper 12c.....	1-15
2 Preparing to Upgrade Fusion Middleware Infrastructure	
2.1 Oracle Fusion Middleware Pre-Upgrade Checklist.....	2-2
2.2 Creating a Complete Backup	2-4
2.2.1 Backing Up the Schema Version Registry Table.....	2-4
2.2.2 Maintaining Custom Domain Environment Settings	2-4
2.3 Cloning Your Production Environment for Testing	2-5
2.4 Verifying Certification and System Requirements	2-6
2.4.1 Verify Your Environment Meets Certification Requirements	2-6

2.4.2	Verify System Requirements and Specifications	2-7
2.4.3	Verify that the Database Hosting Oracle Fusion Middleware Is Supported.....	2-10
2.4.4	Verify that the JDK is Certified for This Release of Oracle Fusion Middleware.....	2-10
2.5	Updating Policy Files when Using Enhanced Encryption (AES 256)	2-10
2.6	Purging Unused Data.....	2-11
2.7	Creating an Edition on the Server for Edition-Based Redefinition	2-11
2.8	Creating a Non-SYSDBA User to Run Upgrade Assistant	2-12

3 Upgrading the Fusion Middleware Infrastructure

3.1	Completing the Pre-Upgrade Tasks for Infrastructure (Required)	3-2
3.1.1	Maintaining Your Custom setDomainEnv Settings (Optional).....	3-3
3.1.2	Using No-Auth SSL Mode in OID Security Store.....	3-4
3.1.3	Removing the Server Instance Scope from OWSM Policy Sets	3-5
3.1.4	Cloning Predefined Documents and Migrating OWSM Policy Attachments	3-5
3.1.5	Running the Readiness Check	3-5
3.2	Installing Fusion Middleware Infrastructure	3-27
3.3	Stopping Servers and Processes	3-29
3.4	Determining Which Schemas to Create.....	3-29
3.5	Creating the Required Schemas with the RCU	3-30
3.6	Using the Schema Version Registry to Identify the Existing Schemas	3-33
3.7	About Upgrading Schemas using the Upgrade Assistant.....	3-33
3.8	Identifying Schemas that Can be Upgraded with the Upgrade Assistant	3-34
3.9	Upgrading the Schemas with the Upgrade Assistant	3-35
3.10	Reconfiguring the Domain with the Reconfiguration Wizard.....	3-37
3.11	Upgrading the Domain Component Configurations with the Upgrade Assistant.....	3-42
3.12	Troubleshooting the Infrastructure Upgrade	3-44
3.12.1	Authentication Failure — JSchException: Auth Fail	3-45
3.12.2	Error while Copying User Messaging Service (UMS) Configuration Files	3-45
3.13	Performing the Post-Upgrade Tasks.....	3-46

4 Tasks to Perform After Upgrade

4.1	Using the Upgrade Validation Checklist.....	4-2
4.2	Starting and Stopping Servers in the Correct Order	4-3
4.2.1	Starting the Node Manager.....	4-4
4.2.2	Starting the Administration Server.....	4-4
4.2.3	Starting the Webtier (Oracle HTTP Server).....	4-5
4.3	Verifying the Domain-specific-Component Configurations Upgrade	4-5
4.4	Reapplying Custom Configuration Settings to setDomainEnv	4-6
4.5	Configuring an Oracle Fusion Middleware 12c Audit Data Store	4-6
4.6	Maintaining the Security Status of Older Java EE Web Service Applications.....	4-6
4.7	Documentation Resources for Managing your Oracle Fusion Middleware 12c Software....	4-7
4.8	Using Your 11g Application Deployments in Oracle Fusion Middleware 12c	4-7
4.8.1	About Oracle Application Development Framework (ADF) 12c.....	4-8

4.8.2	About Oracle JDeveloper 12c.....	4-8
4.9	If Your Existing Environment is a Clustered Configuration.....	4-9
	Packing the Domain on the Primary Node	4-9
	Unpacking the Domain on the Secondary Node	4-9
A	Starting the Upgrade Assistant	
A.1	Starting the Upgrade Assistant in Graphical User Interface (GUI) Mode	A-1
	A.1.1 Starting the Upgrade Assistant with Additional Parameters (Optional).....	A-2
A.2	Starting the Upgrade Assistant in Response File Mode.....	A-5
	A.2.1 Creating an Upgrade Response File	A-5
	A.2.2 Using the Response File to Upgrade Fusion Middleware.....	A-6
A.3	Setting the DISPLAY Environment Variable.....	A-9

Preface

Use this guide to upgrade the schemas, domain configurations, and supported 11g Oracle Fusion Middleware technologies to Oracle Fusion Middleware Infrastructure 12c (12.2.1).

This guide also provides information for upgrading from Oracle Fusion Middleware Infrastructure 12c (12.1.2 or 12.1.3) to 12c (12.2.1).

[Audience](#)

[Documentation Accessibility](#)

[Related Documents](#)

[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

For more information, see the following documents in the Oracle Fusion Middleware 12c (12.2.1) documentation set:

- *Planning an Upgrade of Oracle Fusion Middleware*
- *Installing and Configuring the Oracle Fusion Middleware Infrastructure*
- *Upgrading Oracle WebLogic Server*

- *Upgrading Oracle HTTP Server*
- *Upgrading Oracle Data Integrator*
- *Upgrading Oracle SOA Suite and Business Process Management*

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

The topics in this chapter introduce 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 \(12.2.1.1\)](#)

[Other Changes to This Guide](#)

New and Changed Features for 12c (12.2.1.1)

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.

For more information, see *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.

For more information, see *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.

For more information, see *Upgrading Oracle WebLogic Server*.

Other Changes to This Guide

As of release 12c (12.1.2), there is a new book for Oracle Fusion Middleware documentation library. However, it contains information similar to the Oracle ADF content formerly included in the Oracle Fusion Middleware 11g version of *Oracle Fusion Middleware Upgrade Guide for SOA, WebCenter, and ADF*.

For Oracle Fusion Middleware 12c, a separate guide was required to describe the upgrade process for Oracle Fusion Middleware 11g users who have installed and

configured an Application Developer 11g environment for the deployment and management of their Java and Oracle ADF custom applications.

Introduction to Oracle Fusion Middleware Infrastructure Upgrade

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

Review the topics in this chapter to understand and prepare for the upgrade. This chapter contains the following topics:

[Understanding 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 and the Oracle Fusion Middleware component configurations in that domain to Oracle Fusion Middleware Infrastructure 12c (12.2.1.1).

[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.

[Understanding the Standard Upgrade Topologies for Infrastructure](#)

The standard infrastructure upgrade topology helps you to understand pre-upgrade and post-upgrade environments.

[Understanding the Additional New Features for Oracle Fusion Middleware 12c](#)

Before you begin the upgrade to Oracle Fusion Middleware Infrastructure 12c, review the new features and changes available in Oracle Fusion Middleware 12c.

[Flow Chart and Task Roadmaps for Upgrading to Fusion Middleware Infrastructure](#)

The sections in this topic describe the high-level steps to upgrade the standard topology for Fusion Middleware Infrastructure.

[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.

1.1 Understanding 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 and the Oracle Fusion Middleware component

configurations in that domain to Oracle Fusion Middleware Infrastructure 12c (12.2.1.1).

You can upgrade to this version of Oracle Fusion Middleware Infrastructure from the following supported starting points:

- Oracle Fusion Middleware Application Developer 11g (11.1.1.6, 11.1.1.7, 11.1.1.8, 11.1.1.9)
- Oracle Fusion Middleware Infrastructure 12c (12.1.2.0, 12.1.3.0, 12.2.1.0).

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 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 (12.2.1.1) 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 packs 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.

1.1.1 About Oracle Fusion Middleware Infrastructure 12c

Oracle Fusion Middleware Infrastructure distribution, which is available as part of the Oracle Fusion Middleware 12c (12.2.1.1) 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 packs the WebLogic Server and the Java Required Files (JRF) that are required to install other Fusion Middleware products.

1.1.2 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.

1.1.2.1 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. For more information, see *Understanding and Obtaining the Upgrade and Configuration Tools in Planning an Upgrade of Oracle Fusion Middleware*.

1.1.2.2 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. For more information, 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:	More Information
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. 	<ul style="list-style-type: none"> - Upgrading an OID-Based Security Store
<hr style="border-top: 3px double #000;"/> <p>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.</p> <hr style="border-top: 3px double #000;"/>		
File-based policy store and the OPSS schema in 11g	Use the RCU to create the 12c OPSS schema.	NA.
Database-based policy store and the OPSS schema	Use the Upgrade Assistant to upgrade the existing 11g OPSS schema.	<ul style="list-style-type: none"> - Using the Reconfiguration Wizard to Reconfigure the Domain

- 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. For more information, see the following table:

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

Table 1-2 (Cont.) 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.
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.

1.1.2.3 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.

1.1.2.4 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 [Understanding the Additional New Features for Oracle Fusion Middleware 12c](#).

To upgrade the Oracle HTTP Server, see *Upgrading Oracle HTTP Server*.

1.2 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.x](#).

1.3 Understanding the Standard Upgrade Topologies for Infrastructure

The standard infrastructure upgrade topology helps you to understand pre-upgrade and post-upgrade environments.

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.

As a result, 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 Oracle Fusion Middleware 12c 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.

[Fusion Middleware Infrastructure Standard Upgrade Topology](#)

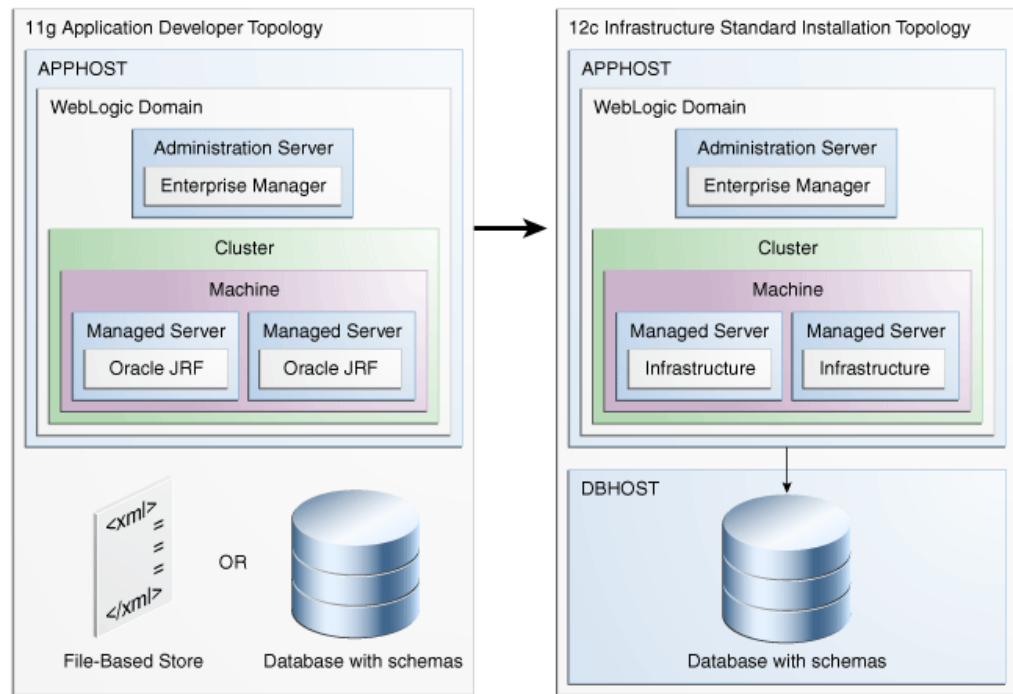
This topic describes the standard topology for upgrading Fusion Middleware Infrastructure to the latest 12c release.

1.3.1 Fusion Middleware Infrastructure Standard Upgrade Topology

This topic describes the standard topology for upgrading Fusion Middleware Infrastructure to the latest 12c release.

[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 Infrastructure Upgrade Topology



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

Table 1-3 Description of the Elements in the 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.</p> <p>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>

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

Element	Description and Links to Additional Documentation
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. For more information, 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>For more information, 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>For more information, see <i>What is the Administration Server</i> in <i>Understanding Oracle Fusion Middleware</i>.</p>
Enterprise Manager	<p>Oracle Enterprise Manager Fusion Middleware Control. This is the main tool that can be used to manage your domain.</p> <p>For more information, see <i>Oracle Enterprise Manager Fusion Middleware Control</i> in <i>Understanding Oracle Fusion Middleware</i>.</p>
Cluster	<p>A collection of multiple WebLogic Server instances running simultaneously and working together.</p> <p>For more information, see <i>Understanding Managed Servers and Managed Server Clusters</i> 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>

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

Element	Description and Links to Additional Documentation
Managed Server	<p>Host for your applications, application components, Web services, and their associated resources.</p> <p>For more information, 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"> • Metadata repository (MDS) <p>This contains metadata for Oracle Fusion Middleware components, such as the Oracle Application Developer Framework.</p> <p>For more information, see What is the Metadata Repository in <i>Understanding Oracle Fusion Middleware</i>.</p> • Oracle Application Developer Framework (ADF) • 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](#).

1.4 Understanding the Additional New Features for Oracle Fusion Middleware 12c

Before you begin the upgrade to Oracle Fusion Middleware Infrastructure 12c, review the new features and changes available in Oracle Fusion Middleware 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?

1.5 Flow Chart and Task Roadmaps for Upgrading to Fusion Middleware Infrastructure

The sections in this topic describe the high-level steps to upgrade the standard topology for Fusion Middleware Infrastructure.

Flow Chart for Upgrading the Infrastructure Standard Upgrade Topologies

This topic helps you to understand the high-level upgrade process with the help of a flowchart.

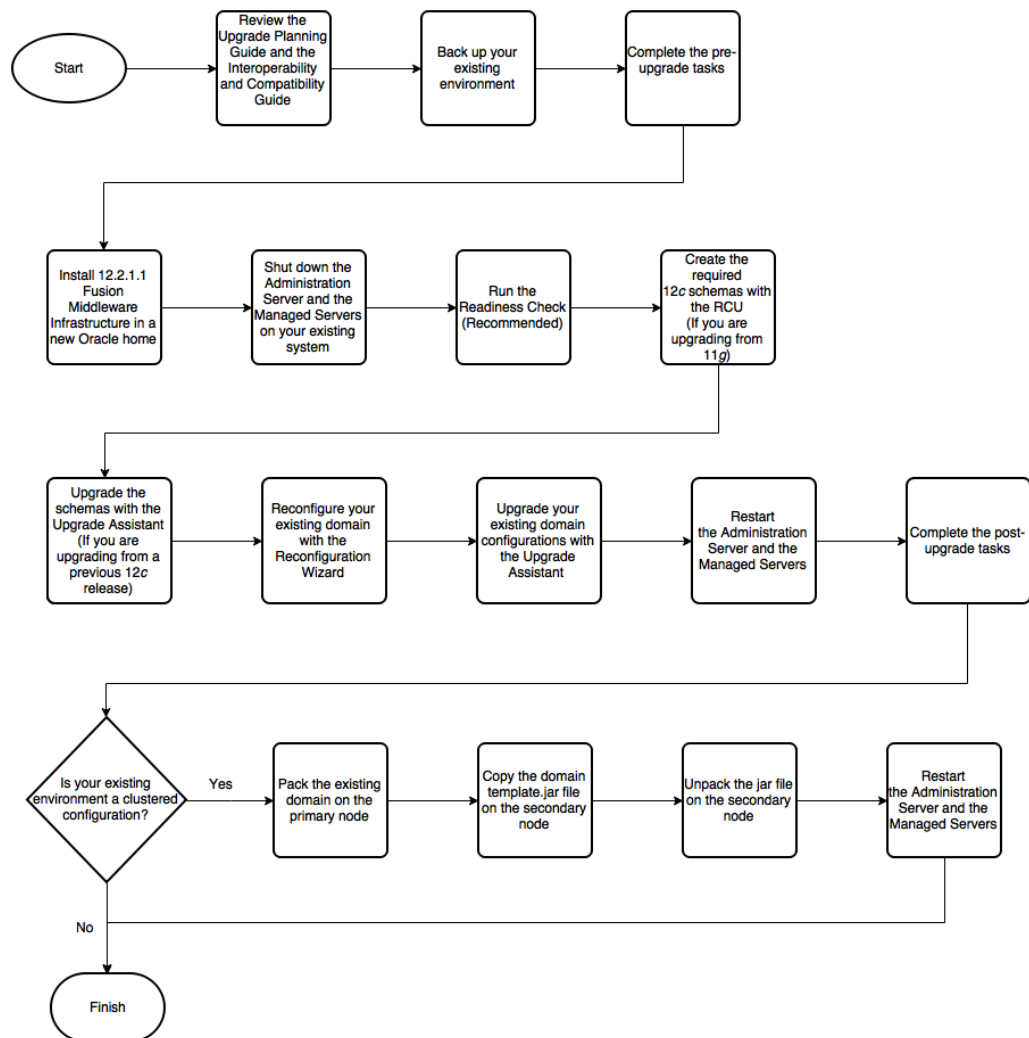
Roadmap for Upgrading the Infrastructure Standard Upgrade Topologies

This topic describes the high-level tasks that you must complete to upgrade to the latest Fusion Middleware Infrastructure 12c release.

1.5.1 Flow Chart for Upgrading the Infrastructure Standard Upgrade Topologies

This topic helps you to understand the high-level upgrade process with the help of a flowchart.

Figure 1-2 Flow Chart for Upgrading Fusion Middleware Infrastructure (Standard Topology)



1.5.2 Roadmap for Upgrading the Infrastructure Standard Upgrade Topologies

This topic describes the high-level tasks that you must complete to upgrade to the latest Fusion Middleware Infrastructure 12c release.

Table 1-4 Oracle Fusion Middleware Infrastructure Upgrade Roadmap

Step No.	Importance (Mandatory, Conditional, Recommended, or Optional)	Task	Description
1	Recommended.	Review the Upgrade Planning Guide and the Interoperability and Compatibility Guide to learn about the interoperability and compatibility factors that could affect how you upgrade to Infrastructure 12.2.1.1.	<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>For planning-related information, see <i>Planning an Upgrade of Oracle Fusion Middleware</i>.</p> <p>To learn about interoperability and compatibility factors, see <i>Understanding Interoperability and Compatibility</i>.</p>
2	Mandatory.	Back up your existing environment.	<p>It is important to back up your existing environment before the upgrade because the domain configurations are upgraded in-place. The back-up allows you to restore your pre-upgrade environment in case the upgrade fails or if the upgrade results are unsatisfactory.</p> <p>For more information, see Creating a Complete Backup.</p>

Table 1-4 (Cont.) Oracle Fusion Middleware Infrastructure Upgrade Roadmap

Step No.	Importance (Mandatory, Conditional, Recommended, or Optional)	Task	Description
3	Recommended.	Complete the pre-upgrade tasks.	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 Oracle Fusion Middleware Pre-Upgrade Checklist .
4	Mandatory.	Install 12.2.1.1 Fusion Middleware Infrastructure in a new Oracle Home.	Install Oracle Fusion Middleware Infrastructure 12c in a new Oracle Home on the host where you installed Oracle Fusion Middleware Application Developer 11g or a previous 12c version of Oracle Fusion Middleware Infrastructure. Follow the procedure described in Installing Fusion Middleware Infrastructure .
5	Mandatory.	Shut down the Administration Server and the Managed Servers in your existing deployment.	Stop the Administration Server and all the Managed Servers as described in Stopping Servers and Processes .
6	Conditional and Recommended: If you are upgrading from a previous 12c release.	Run the Readiness Check.	The Readiness Check helps you to make sure that your pre-upgrade environment and the components within it are ready for upgrade. For more information, see Running the Readiness Check .

Table 1-4 (Cont.) Oracle Fusion Middleware Infrastructure Upgrade Roadmap

Step No.	Importance (Mandatory, Conditional, Recommended, or Optional)	Task	Description
7	Conditional: If you are upgrading from a 11g release.	Create the required 12c schemas with the RCU.	If you are upgrading from 11g release, then you must create the required schemas with the RCU. Unlike Oracle Fusion Middleware 11g, you cannot configure an Oracle Fusion Middleware 12c domain without installing the required schemas in a supported database. For more information, see Creating the Required Schemas with the RCU .
8	Conditional: If you are upgrading from a previous 12c release.	Upgrade the database schemas with the Upgrade Assistant.	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. For more information, see About Upgrading Schemas using the Upgrade Assistant .
9	Mandatory.	Reconfigure your existing domain with the Reconfiguration Wizard.	When you run the Reconfiguration Wizard on your existing domain, it prepares your domain for upgrade by selecting and applying the reconfiguration templates. For more information, see Reconfiguring the Domain with the Reconfiguration Wizard .

Table 1-4 (Cont.) Oracle Fusion Middleware Infrastructure Upgrade Roadmap

Step No.	Importance (Mandatory, Conditional, Recommended, or Optional)	Task	Description
10	Mandatory.	Upgrade the domain configurations with the Upgrade Assistant.	<p>After you have reconfigured your existing domain, you must run the Upgrade Assistant to upgrade any WebLogic component configurations used by your existing domain.</p> <p>For a list of configurations that you can upgrade with the Upgrade Assistant, see Identifying Configurations that can be Upgraded with the Upgrade Assistant.</p>
11	Mandatory.	Restart the servers and 12.2.1.1 instance.	The upgrade process is complete. You can now restart the Administration Server, the Managed Servers, and your 12.2.1.1 instance. See Starting and Stopping Servers in the Correct Order
12	Mandatory.	Complete the post-upgrade tasks.	<p>Perform the post-upgrade tasks that are applicable to your environment. These post-upgrade tasks help you to verify whether your upgrade is successful.</p> <p>For a list of post-upgrade tasks, see Tasks to Perform After Upgrade.</p>
13	Conditional: If your existing environment 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...
14	Conditional: If your existing environment a clustered configuration.	Copy the <code>domaintemplate.jar</code> 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.

Table 1-4 (Cont.) Oracle Fusion Middleware Infrastructure Upgrade Roadmap

Step No.	Importance (Mandatory, Conditional, Recommended, or Optional)	Task	Description
15	Conditional: If your existing environment a clustered configuration.	Unpack the <i>domaintemplate.jar</i> 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.
16	Conditional: If your existing environment a clustered configuration.	Restart the servers and 12.2.1.1 instance.	After you unpack the <i>domaintemplate.jar</i> file on the secondary node, restart the Administration Server, the Managed Servers, and your 12.2.1.1 instance.

1.6 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.

For more information, see *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*.

Preparing to Upgrade Fusion Middleware Infrastructure

Upgrade is performed while the servers are down. The pre-upgrade are often time-consuming. Oracle recommends that you plan and prepare your environment for upgrade by completing these pre-upgrade tasks, so that you have a successful upgrade and a limited downtime.

Use the following checklist to make sure you complete the pre-upgrade tasks:

Oracle Fusion Middleware Pre-Upgrade Checklist

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

Creating a Complete Backup

Before you install the new 12c (12.2.1.1) distributions and begin upgrading your existing Oracle Fusion Middleware deployment, be sure you have backed up all system-critical files; including all of the databases that host your Oracle Fusion Middleware schemas.

Cloning Your Production Environment for Testing

Oracle strongly recommends that you 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

The certification matrix and system requirements documents should be used in conjunction with each other to verify that your environment meets the necessary requirements for installation.

Updating Policy Files when Using Enhanced Encryption (AES 256)

Optional step to perform 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.

[Creating a Non-SYSDBA User to Run Upgrade Assistant](#)

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

2.1 Oracle Fusion Middleware Pre-Upgrade Checklist

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

Upgrades are performed while the servers are down. This checklist 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	Documentation
Create a complete backup of your existing environment.	<p>Required</p> <p>Back up all system-critical files and database(s) that contain any schemas that are to be upgraded before you begin your upgrade.</p> <p>If the upgrade fails, you can restore your pre-upgrade environment and begin the upgrade again.</p>	<p>Creating a Complete Backup.</p> <p>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> <p>Maintaining Custom Domain Environment Settings</p>
Clone your production environment to use as an upgrade testing platform.	<p>Optional</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>Cloning Your Production Environment for Testing .</p>

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

Task	Description	Documentation
<p>Verify that you are installing and upgrading your product on a supported hardware and software configuration.</p> <p>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>Required</p> <p>Verify that your hardware and software configurations (including operating systems) are supported by the latest certifications and requirements documents.</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>You must also make sure to use a supported JDK version before you install the 12c product distributions.</p>	<p>Verifying Certification and System Requirements.</p> <p>If you are currently running a 32-bit operating system, you will have to migrate to a 64-bit operating system before you can upgrade.</p> <p>Migrating from a 32-Bit to a 64-Bit Operating System (Required only if you have a 32-Bit OS)</p>
<p>Update security policy files if you are using enhanced encryption (AES 256).</p> <p>Some of the security algorithms used in Fusion Middleware 12c require additional policy files for the JDK.</p>	<p>Optional</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>Updating Policy Files when Using Enhanced Encryption (AES 256)</p>
<p>Purge any outdated or unused data before you upgrade.</p>	<p>Optional</p> <p>To optimize performance, consider purging data and objects that will not be used in the upgraded environment.</p> <p>Use the purge scripts before you start the instance upgrade to purge the closed 11g instances that you do not need in the upgraded 12c environment.</p>	<p>Purging Unused Data</p>
<p>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 (12.2.1.1).</p>	<p>Required if using an EBR database</p> <p>If you are using an Edition-Based Redefinition (EBR) database, you must create the edition before starting the upgrade.</p>	<p>Creating an Edition on the Server for Edition-Based Redefinition</p>
<p>Create a Non-SYSDBA user to run the Upgrade Assistant.</p>	<p>Optional</p> <p>Oracle recommends that you create the FMW user to run Upgrade Assistant. User FMW can run the Upgrade Assistant without system administration privileges.</p>	<p>Creating a Non-SYSDBA User to Run Upgrade Assistant</p>

2.2 Creating a Complete Backup

Before you install the new 12c (12.2.1.1) distributions and begin upgrading your existing Oracle Fusion Middleware deployment, be sure you have backed up all system-critical files; including all of the databases that host your Oracle Fusion Middleware schemas.

Performing a complete database backup prior to performing a schema upgrade is a prerequisite for running Upgrade Assistant. In the Upgrade Assistant prerequisites GUI screen, you will be required to acknowledge that backups have been performed, before proceeding with the actual upgrade.

For more information, see *Backing Up Your Oracle Fusion Middleware Environment and Upgrading and Preparing Your Oracle Databases for 12c*.

Note: Your system backup must include the `SYSTEM.SCHEMA_VERSION_REGISTRY$` table.

Backing Up the Schema Version Registry Table

Your system backup must include the `SYSTEM.SCHEMA_VERSION_REGISTRY$` table.

Maintaining Custom Domain Environment 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 made to them are overwritten during subsequent domain upgrade and reconfiguration operations.

2.2.1 Backing Up the Schema Version Registry Table

Your system backup must include the `SYSTEM.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. Make sure you back up your existing database schemas before you run the Upgrade Assistant.

Note: Performing these backups prior to performing a schema upgrade is a prerequisite for running Upgrade Assistant. In the Upgrade Assistant prerequisites GUI screen, you will be required to acknowledge that backups have been performed, before proceeding with the actual upgrade.

2.2.2 Maintaining Custom Domain Environment 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 made to them are overwritten during subsequent domain upgrade and reconfiguration operations.

To maintain your custom domain-level environment settings, creating a separate file to store the custom domain information 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 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 `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](#).

2.3 Cloning Your Production Environment for Testing

Oracle strongly recommends that you 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.

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, etc).

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.

2.4 Verifying Certification and System Requirements

The certification matrix and system requirements documents should be used in conjunction with each other 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.

Verify that the JDK is Certified for This Release of Oracle Fusion Middleware

Before you can install any Oracle Fusion Middleware product using a generic installer, you must download and install a supported JDK on your system.

2.4.1 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 proper 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. For more information, see Certification Matrix for 12c (12.2.1.1).

2.4.2 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.

The Oracle Fusion Middleware System Requirements and Specifications document should be used to verify that the requirements of the certification are met. For example, if the certification document indicates that your product is certified for installation on 64-Bit Oracle Linux 7, this document should be used to verify that your Oracle Linux 7 system has met the required minimum specifications, like 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. The latest version is available on Oracle Technology Network.

For a complete description of the system requirements for installing and upgrading to Oracle Fusion Middleware 12c, see Review System Requirements and Specifications.

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 will ensure that the proper owner and group is applied to new Oracle Fusion Middleware 12c files and directories

[Migrating from a 32-Bit to a 64-Bit Operating System \(Required only if you have a 32-Bit OS\)](#)

This step is only required if you are running a 32-bit environment. If you have a 32-bit OS, then you must migrate your 32-bit environment to a 64-bit software environment before you upgrade.

2.4.2.1 Migrating from a 32-Bit to a 64-Bit Operating System (Required only if you have a 32-Bit OS)

This step is only required if you are running a 32-bit environment. If you have a 32-bit OS, then you must migrate your 32-bit environment to a 64-bit software environment before you upgrade.

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

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

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

Upgrading an operating system typically involves the following:

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

[Procure the Hardware That Supports the Upgrade's 64-bit Software Requirement](#)

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

[Stop All Processes](#)

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

[Back Up All Files from the 32-bit Host Machine](#)

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

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

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

[Restore the 11g Backup from 32-bit Host to 64-bit Host](#)

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

[Install the 12c Product Distribution\(s\) on the Target Machine](#)

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

[Upgrade the Target 64-bit Environment Using the Standard Upgrade Procedure](#)

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

2.4.2.1.1 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.

2.4.2.1.2 Stop All Processes

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

For example, to stop the Administration Server, enter the following command:

```
DOMAIN_HOME/bin/stopWebLogic.sh username password [admin_url]
```

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

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

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

For more information on backing up your 11g files, see [Backing Up Your Environment](#) in *Oracle® Fusion Middleware Administrator's Guide*.

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

- 11g Domain Home
- 11g /nodemanager directory located in `$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.

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

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

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

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

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

For detailed information about restoring your 11g files to the 64-bit target machine, see [Recovering Your Environment](#) in *Oracle® Fusion Middleware Administrator's Guide*.

2.4.2.1.6 Install the 12c Product Distribution(s) on the Target Machine

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

For detailed instructions on how to obtain 12c distributions, see [Understanding and Obtaining Product Distributions](#). To identify an installation user see [Selecting an Installation User](#). To understand the directory structure for installation and configuration, see [Understanding Directories for Installation and Configuration](#). Refer to the component-specific installation guides for the component(s) you are installing.

2.4.2.1.7 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.

For a complete upgrade procedure, see the component-specific upgrade guide for the component(s) you are upgrading.

Note: The Node Manager upgrade procedure requires access to the original Node Manager files. Use the 11g 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](#).

2.4.3 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.

It is assumed that you understand the Oracle Database requirements when upgrading and ensure that the database hosting Oracle Fusion Middleware is supported and has sufficient space to perform an upgrade. For more information, see Certification Matrix for 12c (12.2.1.1).

2.4.4 Verify that the JDK is Certified for This Release of Oracle Fusion Middleware

Before you can install any Oracle Fusion Middleware product using a generic installer, you must download and install a supported JDK on your system.

At the time this document was published, the certified JDK was 1.8.0_77.

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

<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*.

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

Optional step to perform 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 require additional policy files for the JDK. For more information, see [Java Cryptography Architecture Oracle Providers Documentation](#).

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.

2.6 Purging Unused Data

Purging unused data and maintaining a purging methodology before an upgrade can optimize the upgrade process.

Note: If a large amount of data needs to be purged, consider partitioning tables or employing other data optimization strategies. Using scripts to remove large amounts of data may impact performance. See *Developing a Purging and Partitioning Methodology* and *Developing a Database Growth Management Strategy*.

Some components have automated purge scripts. If you are using purge scripts, wait until the purge is complete before starting the upgrade process. The upgrade may fail if the purge scripts are running while using the Upgrade Assistant to upgrade your schemas.

For Oracle Data Integrator (ODI) Components:

Purge the execution logs to avoid exporting and importing excessive data as part of work repository export/import in the next step. See *Purging the Logs*.

For SOA Suite Components:

If you are migrating closed 11g instance data, run the instance purge scripts before running the upgrade. See [Using Instance Data Purge Scripts](#).

2.7 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.

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 changes have been made and tested do 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. The new edition for 12c must be a child of your existing 11g or 12c edition.

To create an edition on the database server, log 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_1 as child of
Oracle_FMW_11_1_1_7_0;
```

Where, `Oracle_FMW_11_1_1_7_0` is an example of the edition name you specified in RCU 11.1.1.7 when the 11.1.1.7 schemas were created. Be sure to provide the actual name used when creating the edition.

If the edition is created successfully, you get the following message:

```
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 to manually setup the default edition name for the database, for example:

```
ALTER DATABASE DEFAULT EDITION = Oracle_FMW_12_2_1_1;
```

2.8 Creating a Non-SYSDBA User to Run Upgrade Assistant

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

SYSDBA is an administrative privilege, 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 in this topic must be granted before starting the Upgrade Assistant.

Note: The v\$*xatrans* table does not exist by default. You must run the XAVIEW.SQL script to create this table before creating the user. Moreover, grant on v\$*xatrans* table is required only for Oracle Identity Manager. If you do not require Oracle Identity Manager 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, *welcome1* is the password. Make sure that you specify your actual password when granting privileges.

```
create user FMW identified by welcome1;  
grant dba to FMW;  
grant execute on DBMS_LOB to FMW with grant option;  
grant execute on DBMS_OUTPUT to FMW with grant option;  
grant execute on DBMS_STATS to FMW with grant option;  
grant execute on sys.dbms_aqadm to FMW with grant option;  
grant execute on sys.dbms_aqin to FMW with grant option;  
grant execute on sys.dbms_aqjms to FMW with grant option;  
grant execute on sys.dbms_aq to FMW with grant option;  
grant execute on utl_file to FMW with grant option;  
grant execute on dbms_lock to FMW with grant option;  
grant select on sys.V_$INSTANCE to FMW with grant option;  
grant select on sys.GV_$INSTANCE to FMW with grant option;  
grant select on sys.V_$SESSION to FMW with grant option;  
grant select on sys.GV_$SESSION to FMW with grant option;  
grant select on dba_scheduler_jobs to FMW with grant option;  
grant select on dba_scheduler_job_run_details to FMW with grant option;  
grant select on dba_scheduler_running_jobs to FMW with grant option;  
grant select on dba_aq_agents to FMW with grant option;  
grant execute on sys.DBMS_SHARED_POOL to FMW with grant option;  
grant select on dba_2pc_pending to FMW with grant option;  
grant select on dba_pending_transactions to FMW with grant option;
```

```
grant execute on DBMS_FLASHBACK to FMW with grant option;
grant execute on dbms_crypto to FMW with grant option;
grant execute on DBMS_REPUTIL to FMW with grant option;
grant execute on dbms_job to FMW with grant option;
grant select on pending_trans$ to FMW with grant option;
grant select on dba_scheduler_job_classes to fmw with grant option;
grant select on SYS.DBA_DATA_FILES to FMW with grant option;
grant select on SYS.V_$ASM_DISKGROUP to FMW with grant option;
grant select on v$xsatrans$ to FMW with grant option;
grant execute on sys.dbms_system to FMW with grant option;
grant execute on DBMS_SCHEDULER to FMW with grant option;
grant select on dba_data_files to FMW with grant option;
grant execute on UTL_RAW to FMW with grant option;
grant execute on DBMS_XMLDOM to FMW with grant option;
grant execute on DBMS_APPLICATION_INFO to FMW with grant option;
grant execute on DBMS_UTILITY to FMW with grant option;
grant execute on DBMS_SESSION to FMW with grant option;
grant execute on DBMS_METADATA to FMW with grant option;
grant execute on DBMS_XMLGEN to FMW with grant option;
grant execute on DBMS_DATAPUMP to FMW with grant option;
grant execute on DBMS_MVIEW to FMW with grant option;
grant select on ALL_ENCRYPTED_COLUMNS to FMW with grant option;
grant select on dba_queue_subscribers to FMW with grant option;
grant execute on SYS.DBMS_ASSERT to FMW with grant option;
```

Note:

Oracle Database 11.2.0.3 Database Users ONLY: You must apply **Oracle Patch 13036331** before you begin the upgrade. Go to My Oracle Support to download the patch.

If you do not apply this patch, then you will have to grant additional privileges for some schemas.

Upgrading the Fusion Middleware Infrastructure

This chapter describes the process of upgrading Fusion Middleware Infrastructure to the latest 12c (12.2.1.1) release from a 11g or a previous 12c release.

This chapter contains the following topics:

Completing the Pre-Upgrade Tasks for Infrastructure (Required)

It is important to complete all the standard Oracle Fusion Middleware pre-upgrade tasks associated with your environment before upgrading from Oracle Fusion Middleware 11g Application Developer installation to Oracle Fusion Middleware 12c Infrastructure. Most pre-upgrade tasks are required for all Fusion Middleware components, while some are specific to Infrastructure.

Installing Fusion Middleware Infrastructure

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

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.

Determining Which Schemas to Create

You must create the required schemas depending upon whether your 11g schemas were file-based or database-based.

Creating the Required Schemas with the RCU

If you are upgrading from 11g, you must create the required 12c schemas before you begin the upgrade.

Using the Schema Version Registry to Identify the Existing Schemas

When the schemas are created in your database, RCU creates and maintains a table called `schema_version_registry`. This table contains schema information such as version number, component name and ID, date of creation and modification, and custom prefix. When you run the Upgrade Assistant, it identifies schemas for which an upgrade is available. You can upgrade multiple schemas in a single session of running the Upgrade Assistant.

[About Upgrading Schemas using the Upgrade Assistant](#)

The Upgrade Assistant provides two options for upgrading schemas: **Individually Selected Schemas** and **All Schemas Used By a Domain**.

[Identifying Schemas that Can be Upgraded with the Upgrade Assistant](#)

Review the list of available schemas before you begin the upgrade by querying the schema version registry.

[Upgrading the Schemas with the Upgrade Assistant](#)

Run the Upgrade Assistant to upgrade all the schemas in the existing domain to 12.2.1.1. Oracle recommends that you run the Upgrade Assistant as a non-SYSDBA user.

[Reconfiguring the Domain with the Reconfiguration Wizard](#)

The Reconfiguration Wizard reconfigures the domain while retaining the location of the domain. Use the Reconfiguration Wizard to upgrade your domain to the latest version.

[Upgrading the Domain Component Configurations with the Upgrade Assistant](#)

Follow the instructions in this section to upgrade any additional domain component configurations, such as OWSM policy metadata structure and adapter configurations, using the Upgrade Assistant.

[Troubleshooting the Infrastructure Upgrade](#)

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

[Performing the Post-Upgrade Tasks](#)

After you upgrade Oracle Fusion Middleware 11g Application Developer to Oracle Fusion Middleware 12c Infrastructure, you must complete the post-upgrade tasks.

3.1 Completing the Pre-Upgrade Tasks for Infrastructure (Required)

It is important to complete all the standard Oracle Fusion Middleware pre-upgrade tasks associated with your environment before upgrading from Oracle Fusion Middleware 11g Application Developer installation to Oracle Fusion Middleware 12c Infrastructure. Most pre-upgrade tasks are required for all Fusion Middleware components, while some are specific to Infrastructure.

Caution:

In addition to completing the pre-upgrade tasks required for Infrastructure, you must also complete all of the applicable pre-upgrade tasks described in [Oracle Fusion Middleware Pre-Upgrade Checklist](#).

You must back up your existing environment. If the upgrade fails for any reason, you will have to restart the upgrade process from the source backup.

For more information, see Backup and Recovery Strategies for Upgrade in *Planning an Upgrade of Oracle Fusion Middleware*.

For Infrastructure-specific pre-upgrade tasks, see the following:

[Maintaining Your Custom setDomainEnv Settings \(Optional\)](#)

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.

Running the Readiness Check

Oracle recommends that you run the Readiness Check before starting the upgrade process in order to ensure that your pre-upgrade environment is ready for upgrade.

3.1.1 Maintaining Your Custom `setDomainEnv` Settings (Optional)

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](#).

3.1.2 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.

3.1.3 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.

3.1.4 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*.

3.1.5 Running the Readiness Check

Oracle recommends that you run the Readiness Check before starting the upgrade process in order to ensure that your pre-upgrade environment is ready for upgrade.

[About Running a Pre-Upgrade Readiness Check with the Upgrade Assistant](#)

[Readiness Check Screens](#)

This section describes the screens that are presented when running the Upgrade Assistant in -readiness mode.

3.1.5.1 About Running a Pre-Upgrade Readiness Check with the Upgrade Assistant

You can run the Upgrade Assistant in -readiness mode to detect issues before you perform the actual upgrade. This can be done using the GUI or with silent upgrades using the response files.

The Upgrade Assistant readiness check performs a **read-only**, pre-upgrade review of your existing Oracle Fusion Middleware schemas and Oracle WebLogic configurations.

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.

For more information, see the [Sample Readiness Report](#).

Note: Alternatively, you can run the readiness check in -response mode to perform a silent readiness check using a response file. For more information on using a response file with the Upgrade Assistant, see [Starting the Upgrade Assistant in Response File Mode](#).

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

Readiness checks can be run any number of times before any actual upgrades are attempted. However, do not run the readiness check after an upgrade has been performed, as the report will not provide valid results.

Note:

Oracle recommends that you run the readiness checks during off-peak hours to prevent possible performance degradation.

[Starting the Upgrade Assistant in Readiness Mode](#)

[Performing the Readiness Check](#)

[Understanding the Readiness Report](#)

3.1.5.1.1 Starting the Upgrade Assistant in Readiness Mode

To perform a readiness check on your pre-upgrade environment, you will launch the Upgrade Assistant in `-readiness` mode as shown below:

1. Change directory to `$ORACLE_HOME/oracle_common/upgrade/bin` on UNIX operating systems or `%ORACLE_HOME%/oracle_common/upgrade/bin` on Windows operating systems.
2. Enter the following command to start the Upgrade Assistant.

On UNIX operating systems:

```
./ua -readiness
```

On Windows operating systems:

```
ua.bat -readiness
```

Provide the required information in each of the Upgrade Assistant screens. The screens you see will vary depending on the upgrade options you select. The sections below describe the upgrade options and the information you will need to provide.

3.1.5.1.2 Performing the Readiness Check

When the Upgrade Assistant is started in `-readiness` mode, the following screens appear.

Alternatively, you can run the readiness check using a response file. For more information on using a response file with the Upgrade Assistant, see [Starting the Upgrade Assistant in Response File Mode](#).

Note that these screens are a subset of the screens you will see.

Table 3-1 Upgrade Assistant Screens: Readiness Check

Screen	When Screen Appears	Description
Welcome	Always.	This screen provides an overview of the readiness check.

Table 3-1 (Cont.) Upgrade Assistant Screens: Readiness Check

Screen	When Screen Appears	Description
Readiness Check Type: <ul style="list-style-type: none"> Individually Selected Schemas Domain Based 	Always.	Readiness checks are only performed on schemas or component configurations that are at a supported upgrade starting point. There are two options to choose from. These options are described below: <ul style="list-style-type: none"> Readiness Check Type: Individually Selected Schemas Use the Individually Selected Schemas option to be able to select the schemas you want to review prior to upgrade. Readiness Check Type: Domain Based Use the Domain Based option to let the Upgrade Assistant perform a readiness check per domain.
Available Components	When Individually Selected Schemas option is selected.	This screen lists the available components for which the schemas will be selected. If you select something here, readiness check will be performed on that component's schema.
Schema Credentials	Always.	Use this screen to enter information required to connect to the selected schema and the database that hosts the schema. If the schema that is to be upgraded was created by RCU in a prior Fusion Middleware release then you will see a drop-down menu listing the possible schema names.
Readiness Summary	Always.	This screen provides a high-level overview of the readiness checks to be performed based on your selections. Click Save Response File if you plan to run the Upgrade Assistant again in —response (or silent) mode.

Table 3-1 (Cont.) Upgrade Assistant Screens: Readiness Check

Screen	When Screen Appears	Description
Readiness Check	Always.	<p>This screen displays the current status of the readiness check. Depending on what you have selected to check, the process can take several minutes.</p> <p>For a detailed report, click View Readiness Report. This button appears only after all the readiness checks are complete.</p>
		<hr/> <p>Caution : To prevent performance degradation, consider running the readiness check during off-peak hours.</p> <hr/>
Readiness Success	If the readiness check completes successfully.	<p>You can now review the complete report.</p> <p>If the readiness check encounters an issue or error, review the log file to identify the issues, correct the issues, and then restart the readiness check.</p>

3.1.5.1.3 Understanding the Readiness Report

Now that you have completed the readiness checks for your domain, review the report to determine what actions - if any - need to be taken before the completion of a successful upgrade.

Each Readiness Report contains the following information:

Report Information	Description	Required Action
Overall Readiness Status: SUCCESS or FAILURE	The top of the report indicates whether the Upgrade 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.

Report Information	Description	Required Action
Timestamp	This is the date and time that the report was generated.	No action required.
Log file location	This is the directory location of the generated log file.	No action required.
Readiness Report location	This is the directory location of the generated readiness report.	No action required.
Names of components that were checked	The names and versions of the components included in the check and status.	If your domain includes components that cannot be upgraded to this release, such as SOA Core Extension, then 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, then do not attempt an upgrade.
Status: FAIL	The individual readiness check test detected an issue.	Do not upgrade until all FAILED issues have been resolved.
Status: PASS	The readiness check test detected no issues.	If your readiness check report shows only the PASS status, then 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.

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

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

```
This readiness check report was created on Tue May 30 11:15:52 EDT 2016
Log file is located at: /scratch/yourname/oracle/work/middleware_latest/
oracle_common/upgrade/logs/ua2016-05-30-11-14-06AM.log
Readiness Check Report File: /scratch/yourname/oracle/work/middleware_latest/
oracle_common/upgrade/logs/readiness2016-05-30-11-15-52AM.txt
```

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

```
Oracle Metadata Services
```

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

```
Schema User Name: DEV11_MDS
```

```
Database Type: Oracle Database
```

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

```
VERSION Schema DEV11_MDS is currently at version 12.1.1.1.0. Readiness checks
will now be performed.
```

```
Starting schema test: TEST_REQUIRED_TABLES Test that the schema contains all
the required tables
```

```
Completed schema test: TEST_REQUIRED_TABLES --> Test that the schema contains all
the required tables +++ PASS
```

```
Starting schema test: TEST_REQUIRED_PROCEDURES Test that the schema contains
all the required stored procedures
EXCEPTION Schema is missing a required procedure: GETREPOSITORYFEATURES
Completed schema test: TEST_REQUIRED_PROCEDURES --> Test that the schema contains
all the required stored procedures +++ FAIL
Starting schema test: TEST_REQUIRED_VIEWS Test that the schema contains all the
required database views
Completed schema test: TEST_REQUIRED_VIEWS --> Test that the schema contains all
the required database views +++ PASS
Starting index test for table MDS_ATTRIBUTES: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes
Completed index test for table MDS_ATTRIBUTES: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes +++ PASS
Starting index test for table MDS_COMPONENTS: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes
Completed index test for table MDS_COMPONENTS: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes +++ PASS
Starting index test for table MDS_DEPENDENCIES: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes
Completed index test for table MDS_DEPENDENCIES: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes +++ PASS
Starting index test for table MDS_DEPL_LINEAGES: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes
Completed index test for table MDS_DEPL_LINEAGES: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes +++ PASS
Starting index test for table MDS_LABELS: TEST_REQUIRED_INDEXES --> Test that
the table contains all the required indexes
Completed index test for table MDS_LABELS: TEST_REQUIRED_INDEXES --> Test that
the table contains all the required indexes +++ PASS
Starting index test for table MDS_LARGE_ATTRIBUTES: TEST_REQUIRED_INDEXES -->
Test that the table contains all the required indexes
Completed index test for table MDS_LARGE_ATTRIBUTES: TEST_REQUIRED_INDEXES -->
Test that the table contains all the required indexes +++ PASS
Starting index test for table MDS_METADATA_DOCS: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes
Completed index test for table MDS_METADATA_DOCS: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes +++ PASS
Starting index test for table MDS_NAMESPACES: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes
Completed index test for table MDS_NAMESPACES: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes +++ PASS
Starting index test for table MDS_PARTITIONS: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes
Completed index test for table MDS_PARTITIONS: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes +++ PASS
Starting index test for table MDS_PATHS: TEST_REQUIRED_INDEXES --> Test that the
table contains all the required indexes
Completed index test for table MDS_PATHS: TEST_REQUIRED_INDEXES --> Test that the
table contains all the required indexes +++ PASS
Starting index test for table MDS_PURGE_PATHS: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes
Completed index test for table MDS_PURGE_PATHS: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes +++ PASS
Starting index test for table MDS_SANDBOXES: TEST_REQUIRED_INDEXES --> Test that
the table contains all the required indexes
Completed index test for table MDS_SANDBOXES: TEST_REQUIRED_INDEXES --> Test that
the table contains all the required indexes +++ PASS
Starting index test for table MDS_STREAMED_DOCS: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes
Completed index test for table MDS_STREAMED_DOCS: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes +++ PASS
```

```
Starting index test for table MDS_TRANSACTIONS: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes
Completed index test for table MDS_TRANSACTIONS: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes +++ PASS
Starting index test for table MDS_TXN_LOCKS: 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
Starting index test for table MDS_COMPONENTS: TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes
Completed index test for table MDS_COMPONENTS: TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes +++ PASS
Starting index test for table MDS_DEPENDENCIES: TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes
Completed index test for table MDS_DEPENDENCIES: TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes +++ PASS
Starting index test for table MDS_DEPL_LINEAGES: TEST_UNEXPECTED_INDEXES -->
Test that the table does not contain any unexpected indexes
Completed index test for table MDS_DEPL_LINEAGES: TEST_UNEXPECTED_INDEXES -->
Test that the table does not contain any unexpected indexes +++ PASS
Starting index test for table MDS_LABELS: TEST_UNEXPECTED_INDEXES --> Test that
the table does not contain any unexpected indexes
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
Completed index test for table MDS_LARGE_ATTRIBUTES: TEST_UNEXPECTED_INDEXES -->
Test that the table does not contain any unexpected indexes +++ PASS
Starting index test for table MDS_METADATA_DOCS: TEST_UNEXPECTED_INDEXES -->
Test that the table does not contain any unexpected indexes
Completed index test for table MDS_METADATA_DOCS: TEST_UNEXPECTED_INDEXES -->
Test that the table does not contain any unexpected indexes +++ PASS
Starting index test for table MDS_NAMESPACES: TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes
Completed index test for table MDS_NAMESPACES: TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes +++ PASS
Starting index test for table MDS_PARTITIONS: TEST_UNEXPECTED_INDEXES --> Test
```

```

that the table does not contain any unexpected indexes
  Completed index test for table MDS_PARTITIONS: TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes +++ PASS
  Starting index test for table MDS_PATHS: TEST_UNEXPECTED_INDEXES --> Test that
the table does not contain any unexpected indexes
  Completed index test for table MDS_PATHS: TEST_UNEXPECTED_INDEXES --> Test that
the table does not contain any unexpected indexes +++ PASS
  Starting index test for table MDS_PURGE_PATHS: TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes
  Completed index test for table MDS_PURGE_PATHS: TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes +++ PASS
  Starting index test for table MDS_SANDBOXES: TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes
  Completed index test for table MDS_SANDBOXES: TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes +++ PASS
  Starting index test for table MDS_STREAMED_DOCS: TEST_UNEXPECTED_INDEXES -->
Test that the table does not contain any unexpected indexes
  Completed index test for table MDS_STREAMED_DOCS: TEST_UNEXPECTED_INDEXES -->
Test that the table does not contain any unexpected indexes +++ PASS
  Starting index test for table MDS_TRANSACTIONS: TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes
  Completed index test for table MDS_TRANSACTIONS: TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes +++ PASS
  Starting index test for table MDS_TXN_LOCKS: TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes
  Completed index test for table MDS_TXN_LOCKS: TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes +++ PASS
  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
  Completed datatype test for table MDS_COMPONENTS: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes +++ PASS
  Starting datatype test for table MDS_DEPENDENCIES: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes
  Completed datatype test for table MDS_DEPENDENCIES: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes +++ PASS
  Starting datatype test for table MDS_DEPL_LINEAGES: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes
  Completed datatype test for table MDS_DEPL_LINEAGES: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes +++ PASS
  Starting datatype test for table MDS_LABELS: TEST_COLUMN_DATATYPES_V2 --> Test
that all table columns have the proper datatypes
  Completed datatype test for table MDS_LABELS: TEST_COLUMN_DATATYPES_V2 --> Test
that all table columns have the proper datatypes +++ PASS
  Starting datatype test for table MDS_LARGE_ATTRIBUTES: TEST_COLUMN_DATATYPES_V2
--> Test that all table columns have the proper datatypes
  Completed datatype test for table MDS_LARGE_ATTRIBUTES: TEST_COLUMN_DATATYPES_V2
--> Test that all table columns have the proper datatypes +++ PASS
  Starting datatype test for table MDS_METADATA_DOCS: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes

```

```
Completed datatype test for table MDS_METADATA_DOCS: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes +++ PASS
Starting datatype test for table MDS_NAMESPACES: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes
Completed datatype test for table MDS_NAMESPACES: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes +++ PASS
Starting datatype test for table MDS_PARTITIONS: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes
Completed datatype test for table MDS_PARTITIONS: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes +++ PASS
Starting datatype test for table MDS_PATHS: TEST_COLUMN_DATATYPES_V2 --> Test
that all table columns have the proper datatypes
Completed datatype test for table MDS_PATHS: TEST_COLUMN_DATATYPES_V2 --> Test
that all table columns have the proper datatypes +++ PASS
Starting datatype test for table MDS_PURGE_PATHS: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes
Completed datatype test for table MDS_PURGE_PATHS: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes +++ PASS
Starting datatype test for table MDS_SANDBOXES: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes
Completed datatype test for table MDS_SANDBOXES: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes +++ PASS
Starting datatype test for table MDS_STREAMED_DOCS: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes
Completed datatype test for table MDS_STREAMED_DOCS: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes +++ PASS
Starting datatype test for table MDS_TRANSACTIONS: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes
Completed datatype test for table MDS_TRANSACTIONS: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes +++ PASS
Starting datatype test for table MDS_TXN_LOCKS: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes
Completed datatype test for table MDS_TXN_LOCKS: TEST_COLUMN_DATATYPES_V2 -->
Test that all table columns have the proper datatypes +++ PASS
Starting permissions test for table MDS_ATTRIBUTES: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions
Completed permissions test for table MDS_ATTRIBUTES: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions +++ PASS
Starting permissions test for table MDS_COMPONENTS: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions
Completed permissions test for table MDS_COMPONENTS: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions +++ PASS
Starting permissions test for table MDS_DEPENDENCIES: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions
Completed permissions test for table MDS_DEPENDENCIES: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions +++ PASS
Starting permissions test for table MDS_DEPL_LINEAGES: TEST_TABLE_GRANTS -->
Test that tables have the proper GRANT permissions
Completed permissions test for table MDS_DEPL_LINEAGES: TEST_TABLE_GRANTS -->
Test that tables have the proper GRANT permissions +++ PASS
Starting permissions test for table MDS_LABELS: TEST_TABLE_GRANTS --> Test that
tables have the proper GRANT permissions
Completed permissions test for table MDS_LABELS: TEST_TABLE_GRANTS --> Test that
tables have the proper GRANT permissions +++ PASS
Starting permissions test for table MDS_LARGE_ATTRIBUTES: TEST_TABLE_GRANTS -->
Test that tables have the proper GRANT permissions
Completed permissions test for table MDS_LARGE_ATTRIBUTES: TEST_TABLE_GRANTS -->
Test that tables have the proper GRANT permissions +++ PASS
Starting permissions test for table MDS_METADATA_DOCS: TEST_TABLE_GRANTS -->
Test that tables have the proper GRANT permissions
Completed permissions test for table MDS_METADATA_DOCS: TEST_TABLE_GRANTS -->
```

```

Test that tables have the proper GRANT permissions +++ PASS
  Starting permissions test for table MDS_NAMESPACES: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions
  Completed permissions test for table MDS_NAMESPACES: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions +++ PASS
  Starting permissions test for table MDS_PARTITIONS: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions
  Completed permissions test for table MDS_PARTITIONS: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions +++ PASS
  Starting permissions test for table MDS_PATHS: TEST_TABLE_GRANTS --> Test that
tables have the proper GRANT permissions
  Completed permissions test for table MDS_PATHS: TEST_TABLE_GRANTS --> Test that
tables have the proper GRANT permissions +++ PASS
  Starting permissions test for table MDS_PURGE_PATHS: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions
  Completed permissions test for table MDS_PURGE_PATHS: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions +++ PASS
  Starting permissions test for table MDS_SANDBOXES: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions
  Completed permissions test for table MDS_SANDBOXES: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions +++ PASS
  Starting permissions test for table MDS_STREAMED_DOCS: TEST_TABLE_GRANTS -->
Test that tables have the proper GRANT permissions
  Completed permissions test for table MDS_STREAMED_DOCS: TEST_TABLE_GRANTS -->
Test that tables have the proper GRANT permissions +++ PASS
  Starting permissions test for table MDS_TRANSACTIONS: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions
  Completed permissions test for table MDS_TRANSACTIONS: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions +++ PASS
  Starting permissions test for table MDS_TXN_LOCKS: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions
  Completed permissions test for table MDS_TXN_LOCKS: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions +++ PASS
  Starting permissions test: TEST_DBA_TABLE_GRANTS Test that DBA user has
privilege to view all user tables
  Completed permissions test: TEST_DBA_TABLE_GRANTS --> Test that DBA user has
privilege to view all user tables +++ PASS
  Starting schema test: TEST_ENOUGH_TABLESPACE Test that the schema tablespaces
automatically extend if full
  Completed schema test: TEST_ENOUGH_TABLESPACE --> Test that the schema
tablespaces automatically extend if full +++ PASS
  Starting schema test: TEST_USER_TABLESPACE_QUOTA Test that tablespace quota for
this user is sufficient to perform the upgrade
  Completed schema test: TEST_USER_TABLESPACE_QUOTA --> Test that tablespace quota
for this user is sufficient to perform the upgrade +++ PASS
  Starting schema test: TEST_ONLINE_TABLESPACE Test that schema tablespaces are
online
  Completed schema test: TEST_ONLINE_TABLESPACE --> Test that schema tablespaces
are online +++ PASS
  Starting schema test: TEST_DATABASE_VERSION Test that the database server
version number is supported for upgrade
  INFO Database product version: Oracle Database 11g Enterprise Edition Release
11.2.0.3.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
  Completed schema test: TEST_DATABASE_VERSION --> Test that the database server
version number is supported for upgrade +++ PASS
  Finished readiness check of Oracle Metadata Services with status: FAILURE.

Common Infrastructure Services
  Starting readiness check of Common Infrastructure Services.
  Schema User Name: DEV1212_STB

```

```
Database Type: Oracle Database
Database Connect String: machinename@yourcompany.com
VERSION Schema STB is currently at version 12.1.2.0.0. 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
Completed schema test: TEST_REQUIRED_TABLES --> Test that the schema contains all
the required tables +++ PASS
Starting schema test: TEST_UNEXPECTED_TABLES Test that the schema does not
contain any unexpected tables
Completed schema test: TEST_UNEXPECTED_TABLES --> Test that the schema does not
contain any unexpected tables +++ PASS
Starting schema test: TEST_REQUIRED_VIEWS Test that the schema contains all the
required database views
Completed schema test: TEST_REQUIRED_VIEWS --> Test that the schema contains all
the required database views +++ PASS
Starting schema test: TEST_MISSING_COLUMNS Test that tables and views are not
missing any required columns
Completed schema test: TEST_MISSING_COLUMNS --> Test that tables and views are
not missing any required columns +++ PASS
Starting schema test: TEST_UNEXPECTED_COLUMNS Test that tables and views do not
contain any unexpected columns
Completed schema test: TEST_UNEXPECTED_COLUMNS --> Test that tables and views do
not contain any unexpected columns +++ PASS
Starting 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 permissions test: TEST_DBA_TABLE_GRANTS Test that DBA user has
privilege to view all user tables
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
Completed permissions test: TEST_DBA_TABLE_GRANTS --> Test that DBA user has
privilege to view all user tables +++ PASS
Starting permissions test for table COMPONENT_SCHEMA_INFO: TEST_TABLE_GRANTS -->
Test that tables have the proper GRANT permissions
Completed permissions test for table TEST_TABLE_GRANTS: Test that tables have the
proper GRANT permissions --> PASS +++ {3}
Starting permissions test for table SERVICETABLE: TEST_TABLE_GRANTS --> Test
that tables have the proper GRANT permissions
Completed permissions test for table TEST_TABLE_GRANTS: Test that tables have the
proper GRANT permissions --> PASS +++ {3}
Starting datatype test for table COMPONENT_SCHEMA_INFO: TEST_COLUMN_DATATYPES_V2
--> Test that all table columns have the proper datatypes
Completed datatype test for table COMPONENT_SCHEMA_INFO: TEST_COLUMN_DATATYPES_V2
--> Test that all table columns have the proper datatypes +++ PASS
Starting datatype test for table SERVICETABLE: TEST_COLUMN_DATATYPES_V2 --> Test
that all table columns have the proper datatypes
Completed datatype test for table SERVICETABLE: TEST_COLUMN_DATATYPES_V2 --> Test
that all table columns have the proper datatypes +++ PASS
Starting index test for table COMPONENT_SCHEMA_INFO: TEST_REQUIRED_INDEXES -->
Test that the table contains all the required indexes
Completed index test for table COMPONENT_SCHEMA_INFO: TEST_REQUIRED_INDEXES -->
Test that the table contains all the required indexes +++ PASS
Starting index test for table COMPONENT_SCHEMA_INFO: TEST_UNEXPECTED_INDEXES -->
Test that the table does not contain any unexpected indexes
Completed index test for table COMPONENT_SCHEMA_INFO: TEST_UNEXPECTED_INDEXES -->
```

```

Test that the table does not contain any unexpected indexes +++ PASS
  Starting index test for table SERVICETABLE: TEST_REQUIRED_INDEXES --> Test that
the table contains all the required indexes
  Completed index test for table SERVICETABLE: TEST_REQUIRED_INDEXES --> Test that
the table contains all the required indexes +++ PASS
  Starting index test for table SERVICETABLE: TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes
  Completed index test for table SERVICETABLE: TEST_UNEXPECTED_INDEXES --> Test
that the table does not contain any unexpected indexes +++ PASS
  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_ENOUGH_TABLESPACE Test that the schema tablespaces
automatically extend if full
  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
  Completed schema test: TEST_ENOUGH_TABLESPACE --> Test that the schema
tablespaces automatically extend if full +++ PASS
  Starting schema test: TEST_ONLINE_TABLESPACE Test that schema tablespaces are
online
  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
  Completed schema test: TEST_ONLINE_TABLESPACE --> Test that schema tablespaces
are online +++ PASS
  Starting schema test: TEST_USER_TABLESPACE_QUOTA Test that tablespace quota for
this user is sufficient to perform the upgrade
  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
  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_DATABASE_VERSION Test that the database server
version number is supported for upgrade
  Starting schema test: TEST_DATABASE_VERSION Test that the database server
version number is supported for upgrade
  INFO Database product version: Oracle Database 11g Enterprise Edition Release
11.2.0.3.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
  Completed schema test: TEST_DATABASE_VERSION --> Test that the database server
version number is supported for upgrade +++ PASS
  Completed schema test: TEST_DATABASE_VERSION --> Test that the database server
version number is supported for upgrade +++ PASS
  Finished readiness check of Common Infrastructure Services with status: SUCCESS.

Finished readiness check of components.

```

3.1.5.2 Readiness Check Screens

This section describes the screens that are presented when running the Upgrade Assistant in -readiness mode.

The Upgrade Assistant can be run in -readiness mode before you perform the actual upgrade to detect any potential problems with the pre-upgrade environment.

[Welcome](#)[Readiness Check Type: Individually Selected Schemas](#)[Readiness Check Type: Domain Based](#)

The **Domain Based** option is used to check all of the upgrade-eligible schemas and/or component configurations used by the domain. The Upgrade Assistant detects all of the schemas for you. You can check schemas and component configurations at the same time. Or, if you prefer, you can select one or the other. In either case, you must specify the **Domain Directory** that is to be reviewed.

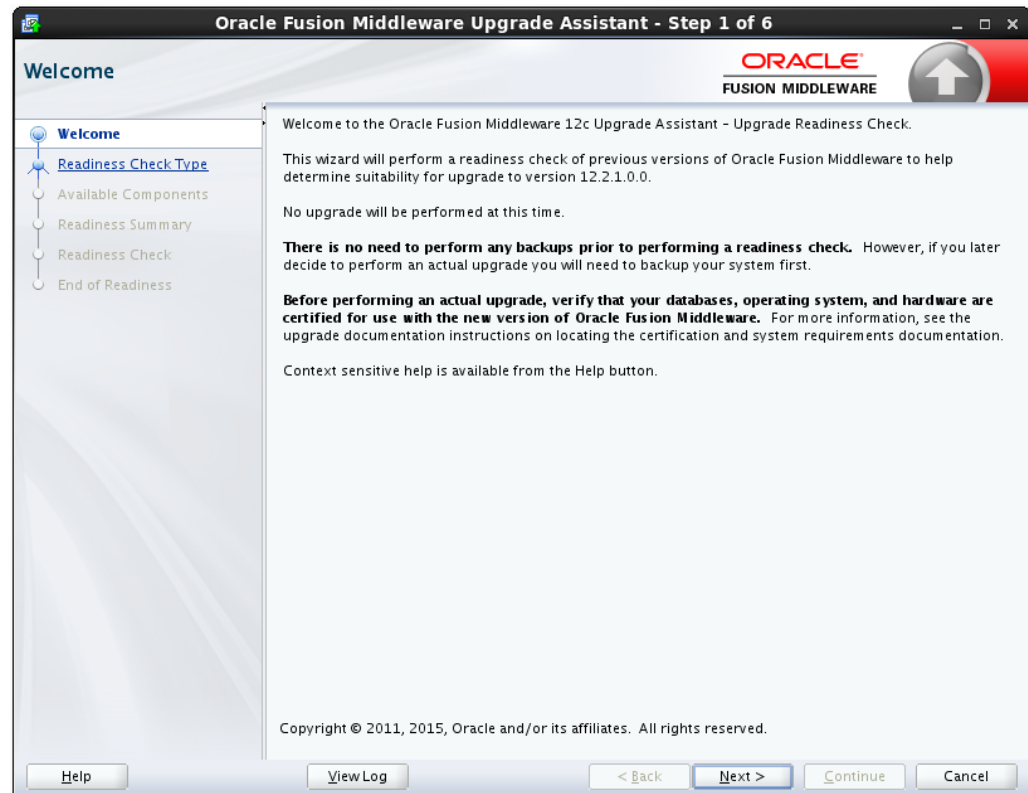
[Available Components](#)

This screen appears if you select **Individually Selected Schemas** in the Schemas screen.

[Schema Credentials](#)[Readiness Summary](#)[Readiness Check](#)[Log Viewer](#)[Readiness Success](#)[Sample Readiness Report](#)

3.1.5.2.1 Welcome

Figure 3-1 *Readiness Welcome*



The Upgrade Assistant Readiness Check performs a read-only, pre-upgrade review of your existing Oracle Fusion Middleware schemas and Oracle WebLogic component configurations.

The Readiness Check provides a formatted, time-stamped Readiness Report so you can address any issues before you attempt the actual upgrade. If no issues are detected, you can begin the upgrade process.

The Upgrade Assistant Readiness Check can be run with your existing 11g or 12c domain online or offline.

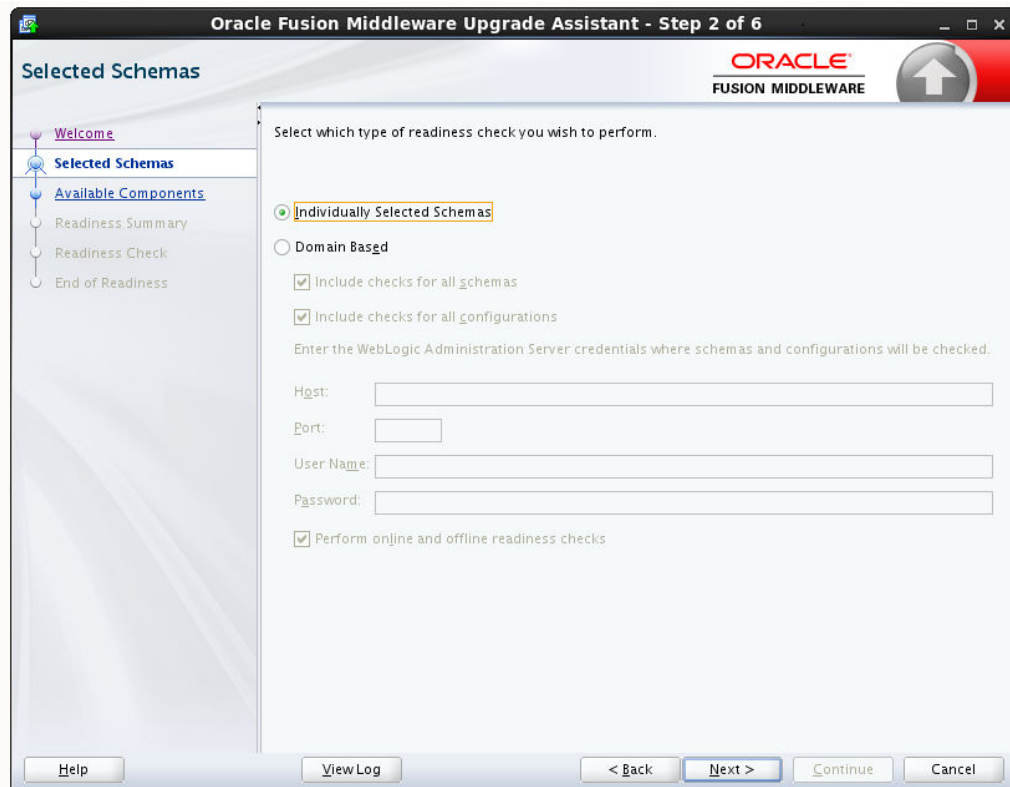
Note: While readiness check ships with 12.2.1, it only checks supported pre-upgrade environments.

The Readiness Check can be run any number of times before the actual upgrade is performed. However, do not run after the Readiness Check after an upgrade has been performed, as the report will not provide valid results.

Oracle recommends that you read this report thoroughly before performing an upgrade.

3.1.5.2.2 Readiness Check Type: Individually Selected Schemas

Figure 3-2 *Individually Selected Schemas*



You have two options when running the readiness check:

- **Individually Selected Schemas**
- **Domain Based**

Select the **Individually Selected Schemas** option to limit the check to specific schemas. Click **Next** and you will be required to supply the schema credentials.

Readiness checks are performed on the schemas that you connect to. The readiness check report tells you whether a schema is supported for an upgrade, or where an upgrade is needed.

3.1.5.2.3 Readiness Check Type: Domain Based

The **Domain Based** option is used to check all of the upgrade-eligible schemas and/or component configurations used by the domain. The Upgrade Assistant detects all of the schemas for you. You can check schemas and component configurations at the same time. Or, if you prefer, you can select one or the other. In either case, you must specify the **Domain Directory** that is to be reviewed.

You have several options when checking the WebLogic Server domain.

You can select one - or more - of the following options each time you run the Domain Based Readiness Check:

- **Include checks for all schemas**

Select this option to enable the Upgrade Assistant to discover and review all components that have a schema available to upgrade.

- **Include checks for all configurations**

Select this option to review component configurations for a managed WebLogic Server domain.

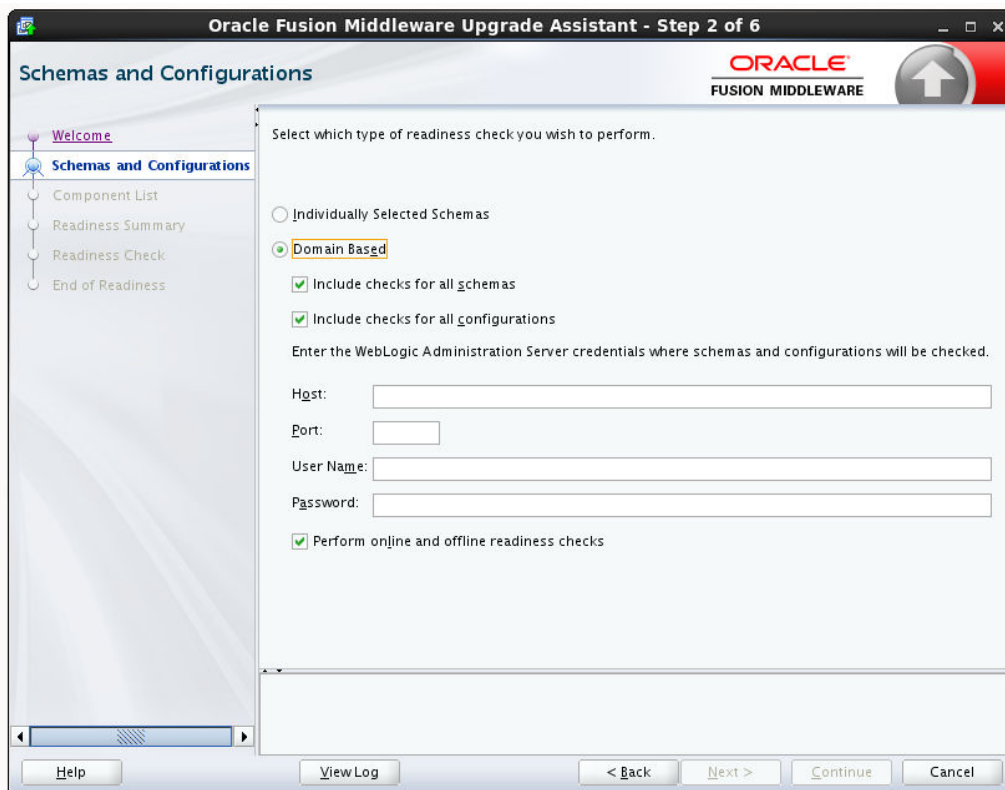
You can perform domain configuration check even when the domain is online or offline.

- **Perform online and offline readiness checks.**

Select this option to perform additional online readiness checks. This option will require your domain to be online. You must provide the domain's host name, port, user name, and password that you plan to check.

If you do not select this option your domain can be offline. You must provide the domain location that you plan to check.

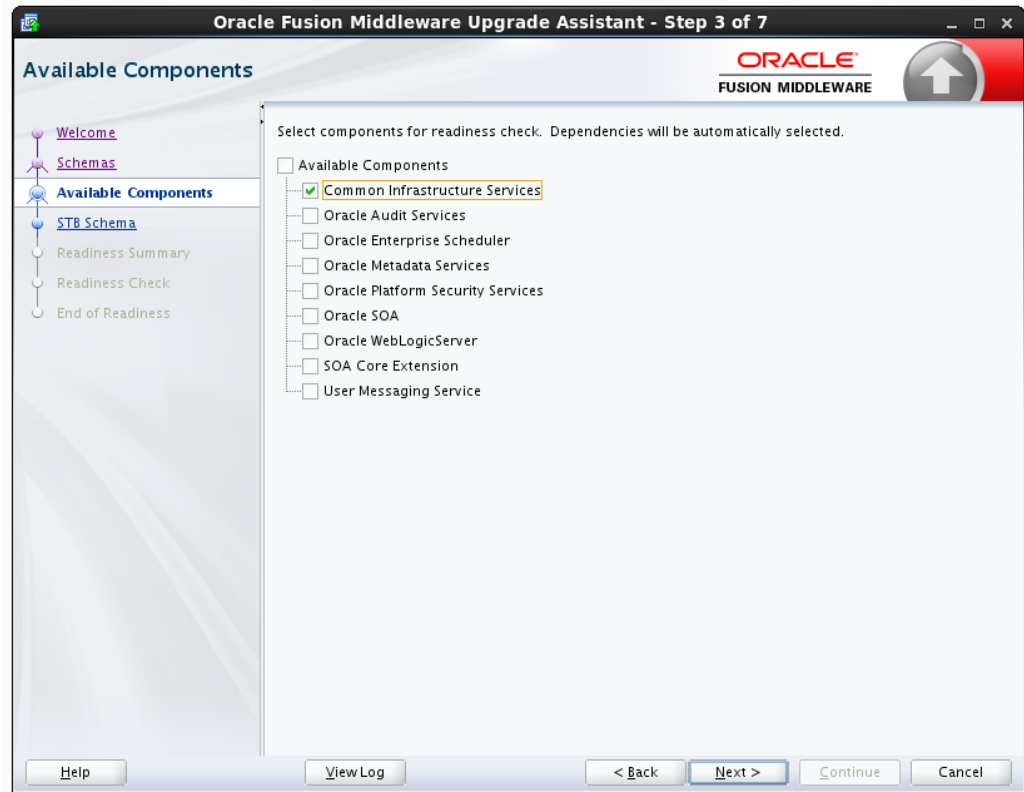
Figure 3-3 WebLogic Server Readiness Check Options



3.1.5.2.4 Available Components

This screen appears if you select **Individually Selected Schemas** in the Schemas screen.

Figure 3-4 Available Components



If you chose **Individually Selected Schemas** this screen lists the available components for which the schemas will be selected. If you select something here, readiness check will be performed on that component's schema. You must select one or more components from the list to perform readiness check on them.

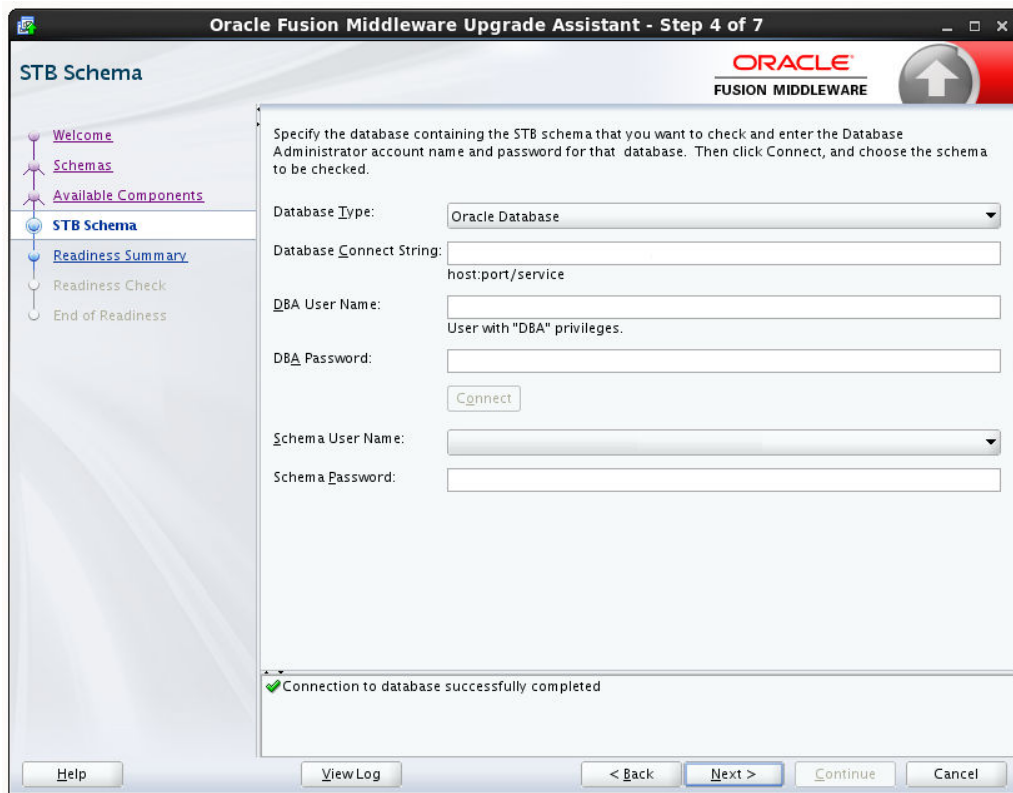
3.1.5.2.5 Schema Credentials

Use this screen to enter information required to connect to the selected schema and the database that hosts the schema. If the schema that is to be reviewed was created by RCU in a prior Fusion Middleware release then you will see a drop-down menu listing the possible schema names as shown below.

Click **Connect** to connect to the database then select the schema to be reviewed. NOTE: Most schemas will have this information pre-populated. If, however, the Upgrade Assistant is unable to detect the connection details, then they must be entered manually as shown below.

If multiple components are selected, then the Schema Credential screens appear in dependency order.

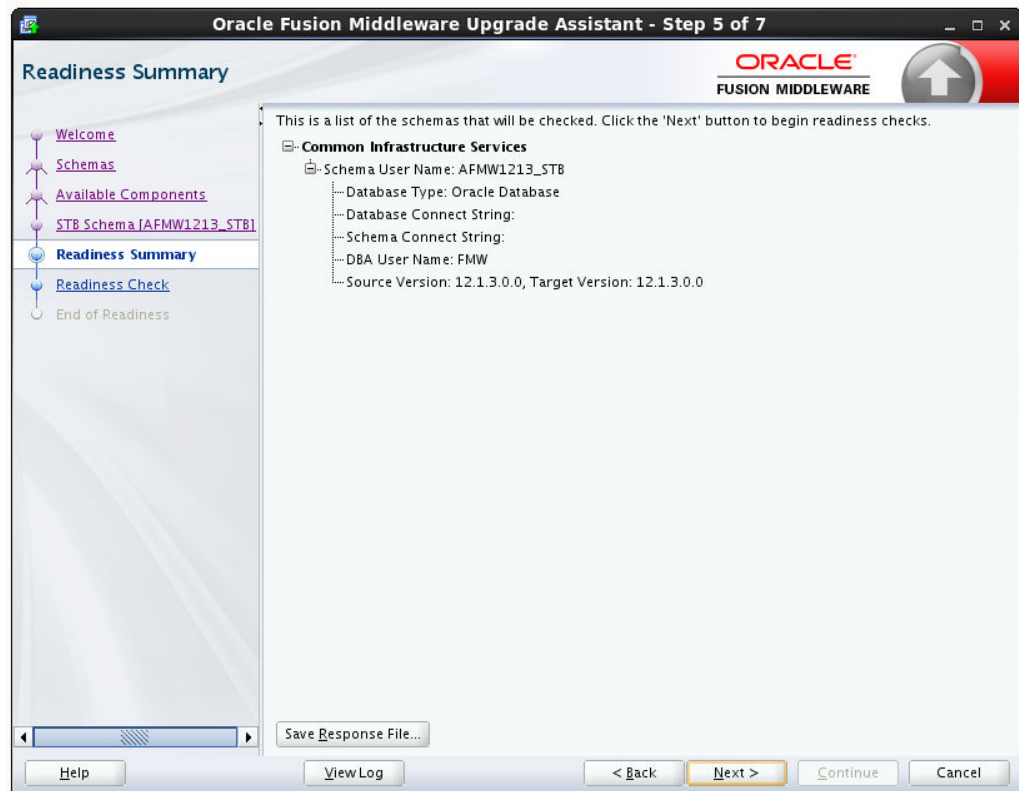
Figure 3-5 Schema Credentials



3.1.5.2.6 Readiness Summary

This screen provides a high-level overview of the readiness checks performed based on your selections.

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

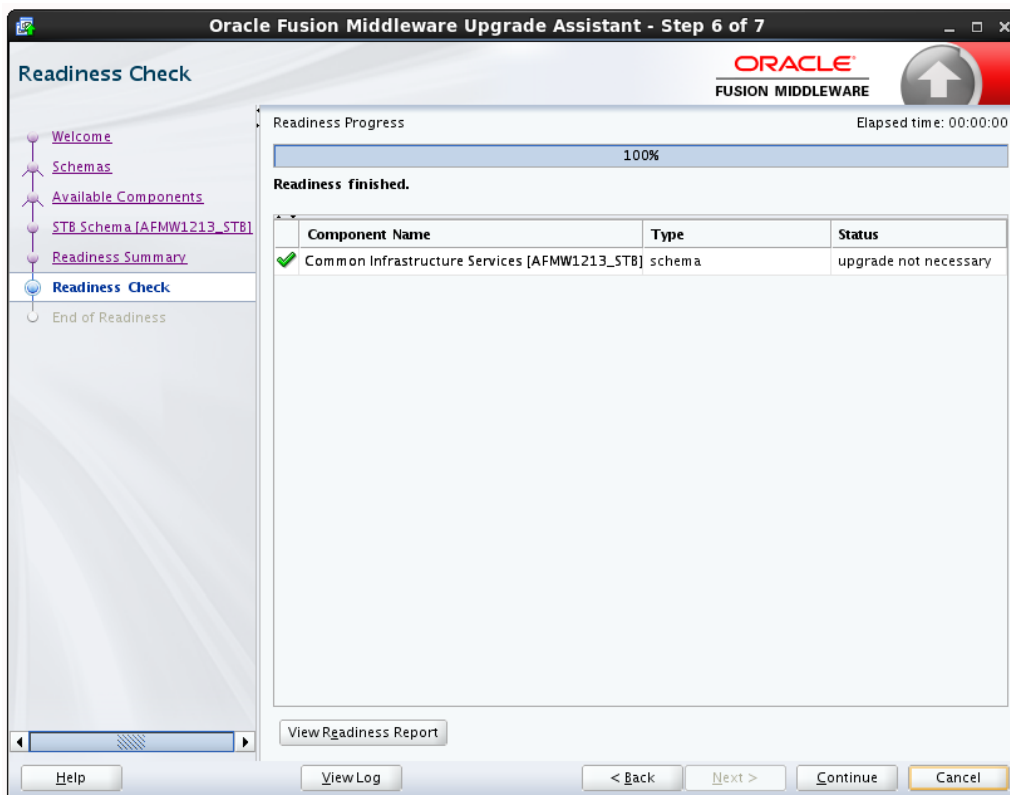
Figure 3-6 Readiness Summary

3.1.5.2.7 Readiness Check

This screen shows the overall progress and completion details of the running readiness check. If you are checking multiple components, then each gets component will have its own progress bar and will be checked in parallel. Once completed, click **View Readiness Report** to see the full text report.

CAUTION: If you are running the readiness check on your online production environment, Oracle recommends that you perform the check during off-peak hours to prevent performance degradation.

Figure 3-7 Readiness Check Complete

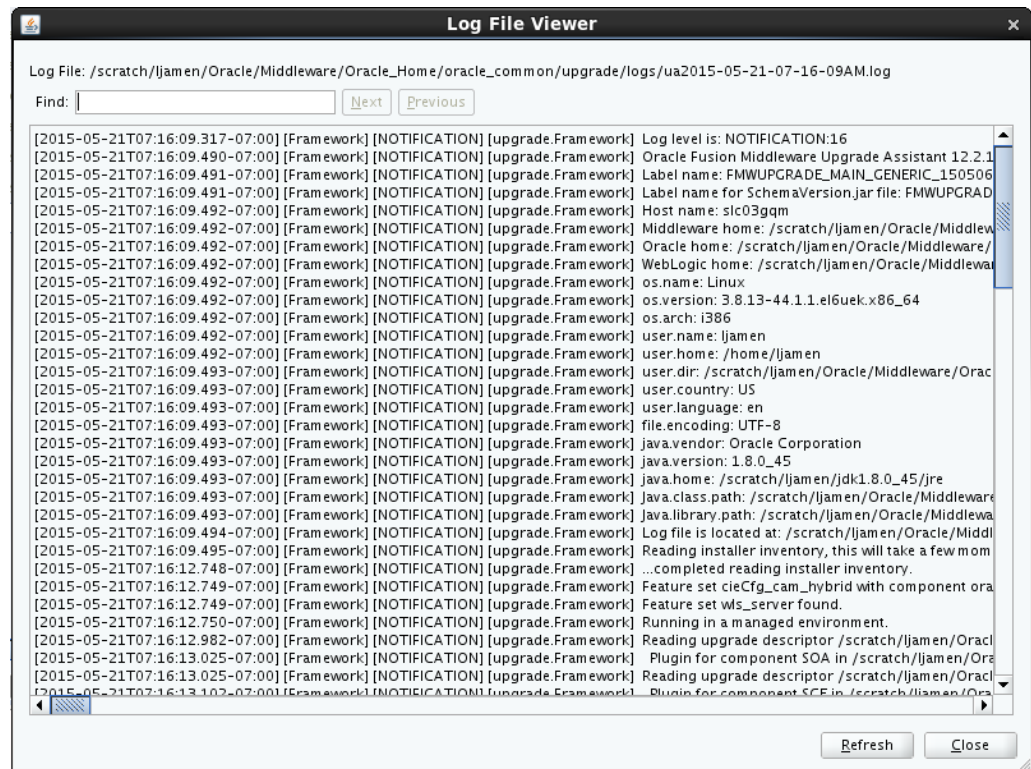


3.1.5.2.8 Log Viewer

Click **View Log** from any of the screens to see the latest logged information.

The log file is managed by the command line options you set. See [Starting the Upgrade Assistant with Additional Parameters \(Optional\)](#) for more information.

Figure 3-8 Sample Log File

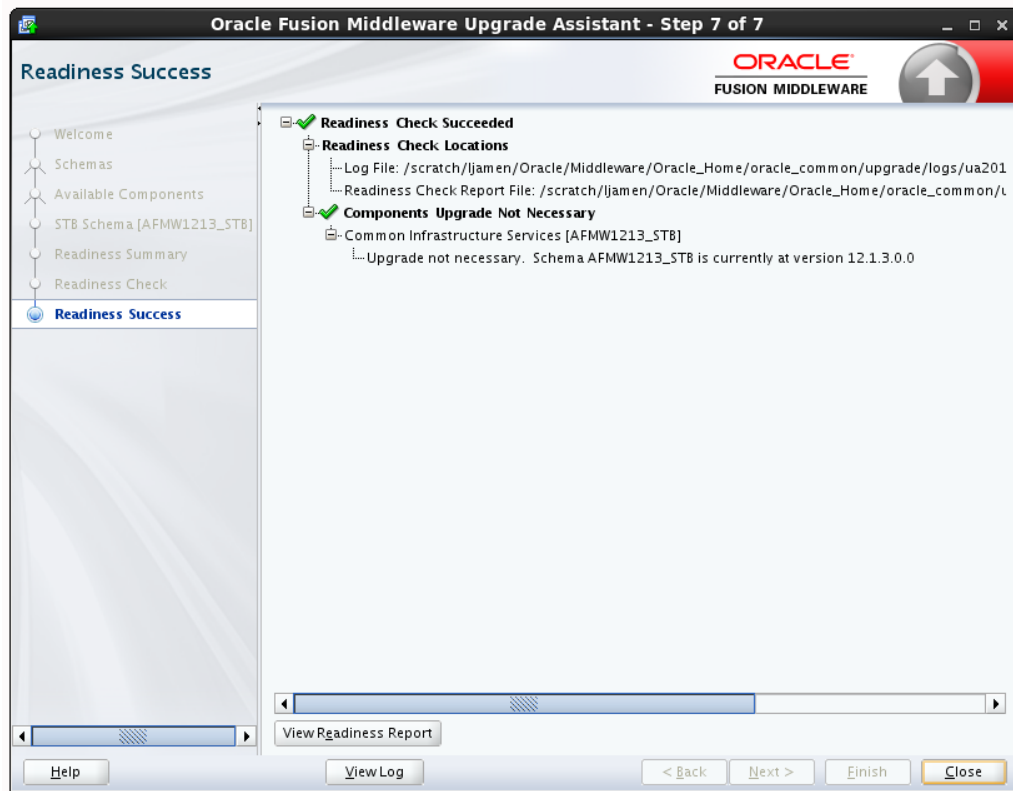


3.1.5.2.9 Readiness Success

Readiness success indicates that the readiness review was successfully completed.

Even with a successful completion of the review, you should still click **View Readiness Report** and review the Readiness Report before you perform the actual upgrade.

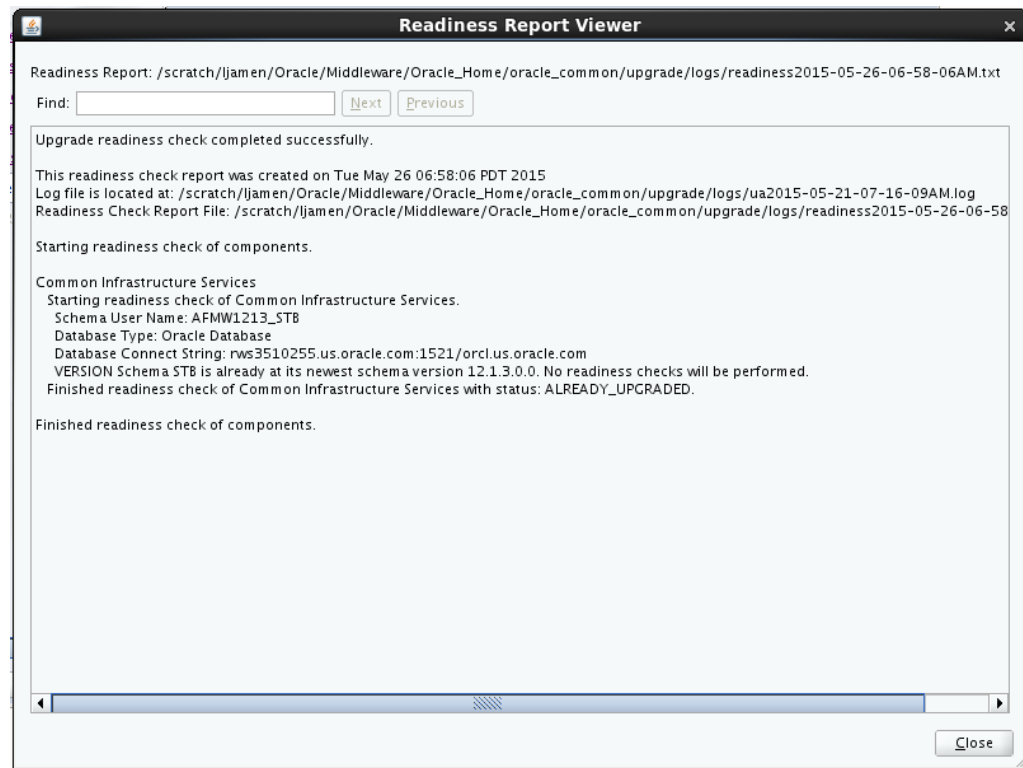
Figure 3-9 End of Readiness: Readiness Success



3.1.5.2.10 Sample Readiness Report

A formatted Readiness Report is prepared for you after running the check. Make sure that you review the report and correct any issues before you start the actual upgrade. Use the Find option to search for a particular word within the report (such as a schema name or command, for example.)

The report also indicates where the completed Readiness Check Report file is located.

Figure 3-10 Readiness Report Viewer

3.2 Installing Fusion Middleware Infrastructure

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

To install Fusion Middleware Infrastructure distribution:

1. Sign in to the target system where you want to install the 12.2.1.1 product distribution.
2. Download the Oracle Fusion Middleware Infrastructure distribution (fmw_12.2.1.1.0_infrastructure_generic.jar) from [Oracle Technology Network](#) or [Oracle Software Delivery Cloud](#) on your target system.
3. Change to the directory where you downloaded the 12.2.1.1 product distribution.
4. Start the installation program by entering the following command:

On UNIX operating system:

```
$JDK_HOME/bin/java [-d64] -jar
fmw_12.2.1.1.0_infrastructure_generic.jar
```

Note: Use the "-d64" flag only if you are using the HP-UX Itanium system.

On Windows operating system:

```
%JDK_HOME%\bin\java -jar
fmw_12.2.1.1.0_infrastructure_generic.jar
```

5. On UNIX operating system, 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: Installation Inventory Setup screen does not appear on Windows operating system.

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

7. On the Auto Updates screen, select **Skip Auto Updates** and click **Next**.

- 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.

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 *Selecting Directories for Installation and Configuration in Planning an Installation of Oracle Fusion Middleware*.

9. On the Installation Type screen, select **Fusion Middleware Infrastructure** and click **Next**.

Note: The topology in this document does not include server examples. Oracle strongly recommends that you do not install examples into a production environment.

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

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**, however this approach is 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 you selected.

To save these options to a response file, click **Save Response File** and enter the location and the name of the response file. You can use response files for silent installation. Click **Install**.

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

3.3 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*.

3.4 Determining Which Schemas to Create

You must create the required schemas depending upon whether your 11g schemas were file-based or database-based.

Consider the following scenarios:

- If you did not use a database in 11g, then you must install and configure a supported database, and also create one or more of the database schemas as described in [Database Schema Requirement for Infrastructure 12c](#).
- If you were already using a database to host the schemas for your Application Developer 11g domain, then use the schema version registry to list the Oracle Fusion Middleware 11g schemas that are already available in your database, as described in [Using the Schema Version Registry to Identify the Existing Schemas](#).

You need not create the schemas listed in the Schema Version Registry manually. Instead, you can later use the Upgrade Assistant to upgrade the 11g schemas during the upgrade process.

However, you must still create the required schemas, as described in [Database Schema Requirement for Infrastructure 12c](#).

Note:

As of release 12c, the Service Table (_STB) schema is required for all Infrastructure installations. The Service Table schema must be upgraded each time the infrastructure is upgraded. You cannot use an older version schema with a new Infrastructure installation.

If you are upgrading from a previous 12c release, then you need not create the 12c schemas. you can directly upgrade them with the Upgrade Assistant.

3.5 Creating the Required Schemas with the RCU

If you are upgrading from 11g, you must create the required 12c schemas before you begin the upgrade.

Note: This procedure assumes that you are a SYS or SYSDBA user with full database administrator privileges. If you are a user with limited database privileges, follow the procedure stated in *Creating Schemas as a User With Limited Database Privileges*. For in-depth information about using RCU, see *Creating Schemas with the Repository Creation Utility*.

If you are upgrading from 11g release, then you must provide the same prefix while creating the 12c schemas.

To create the 12c schema:

1. Change directory to the following:

On UNIX operating system:

```
$ORACLE_HOME/oracle_common/bin
```

On Windows operating system:

```
%ORACLE_HOME%\oracle_common\bin
```

2. Run the RCU by entering the following command:

On UNIX operating system:

```
./rcu
```

On Windows operating system:

```
rcu.bat
```

3. On the Welcome screen, click **Next**.
4. On the Create Repository screen, select **Create Repository** and then select **System Load and Product Load**. Click **Next**.

If you do not have DBA privileges, select **Prepare Scripts for System Load**.

5. On the Database Connection Details screen, select the **Database Type** and enter the following details:

Table 3-2 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.exampdomain.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.exampdomain.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 3-3 Connection Credentials for MySQL Databases

Option	Description and Example
Host Name	Specify the host name, IP address, or complete server name in host\server 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 administrator privileges.
Password	Enter the password for your database user.

Table 3-4 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 host\server 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 3-4 (Cont.) Connection Credentials for Microsoft SQL Server Databases

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

Table 3-5 Connection Credentials for IBM DB2 Databases

Option	Description and Example
Server Name	Specify the host name, IP address, or complete server name in host\server 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 db2admin.
Password	Enter the password for your database user.

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

- On the Select Components screen, select **Create new prefix** and enter the same prefix as the 11g schema.

Select **AS Common Schemas**. All of the schemas in this section are automatically selected. Click **Next**.

The Checking Prerequisites box appears that indicates the progress of the components prerequisites. Click **OK** the operation is complete.

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

You must remember the passwords you enter on this screen; you need this information while configuring your product installation. Oracle recommends that you note these values.

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

When you click **Next**, a separate dialog window appears asking you to confirm that you want to create these tablespaces. Click **OK** to proceed and dismiss the dialog window.

A second dialog window appears showing the progress of tablespace creation. After this is complete, click **OK** to dismiss this window and go to the next screen.

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 RCU. Select the **Encrypt Tablespace** check box on the Map Tablespaces screen to encrypt all new tablespaces that RCU creates.

9. 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. You can click on the name of a particular log file to view the contents of that file.

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

[Determining Which Schemas to Create](#)

[Understanding System Load and Product Load](#)

[Understanding Custom Prefixes](#)

[Planning Your Schema Creation](#)

3.6 Using the Schema Version Registry to Identify the Existing Schemas

When the schemas are created in your database, RCU creates and maintains a table called `schema_version_registry`. This table contains schema information such as version number, component name and ID, date of creation and modification, and custom prefix. When you run the Upgrade Assistant, it identifies schemas for which an upgrade is available. You can upgrade multiple schemas in a single session of running the Upgrade Assistant.

To determine which of your 11g or 12c schemas can be upgraded to 12.2.1.1, see *Identifying Schemas that Can be Upgraded with the Upgrade Assistant* in *Upgrading with the Upgrade Assistant*.

Alternatively, use the following command to query the `schema_version_registry` table to identify the existing schemas:

```
select * from schema_version_registry;
```

3.7 About Upgrading Schemas using the Upgrade Assistant

The Upgrade Assistant provides two options for upgrading schemas: **Individually Selected Schemas** and **All Schemas Used By a Domain**.

Individually Selected Schemas

This option enables you to choose which component schemas to upgrade. Select this option when you want to select individual schemas for upgrade and you do not want to upgrade all of the schemas used by the domain.

For example, if you want to make a trial run of Upgrade Assistant by creating schemas with RCU that are outside the domain, and then use Upgrade Assistant to upgrade them.

All Schemas Used By a Domain

This option allows the Upgrade Assistant to detect all of the available schemas within the specified domain and to include them in the upgrade. This is also known as a domain assisted schema upgrade.

3.8 Identifying Schemas that Can be Upgraded with the Upgrade Assistant

Review the list of available schemas before you begin the upgrade by querying the schema version registry.

This optional step can be used if you want to manually query the `schema_version_registry` table before you start the upgrade process. It is important to note that the Upgrade Assistant identifies all schemas that are available for an upgrade and allows you to select the individual schemas you want to upgrade or allow Upgrade Assistant to upgrade all of the schemas in the domain. In addition, when you run the Upgrade Assistant in `-readiness` mode, you will receive a report with all of the schemas and their current pre-upgrade versions.

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 ;
```

The following report is generated when saved to a SQL script, for example `version.sql`.

If the number in the "VERSION" is at 11.1.1.7.0 or higher, and the STATUS column is 'VALID', then the schema is supported for upgrade.

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

Tip:

Compare the information you collect from the schema version registry and the corresponding schemas to determine whether there are schemas in your domain that are not available for an upgrade yet.

Notes about the schemas that need to be upgraded

- For **most** components, the only schema version starting points that are valid for upgrading are 11g Release 1 (1.1.1.7.0, 11.1.1.8.0, or 11.1.1.9.0) or 12c (12.1.2, 12.1.3, or 12.2.1.0). If your schemas are not at a supported version, then you must upgrade them before using the 12c (12.2.1.1) upgrade procedures.

Some components, such as Oracle Enterprise Data Quality and Oracle Golden Gate Veridata, support an upgrade from versions other than the standard Oracle Fusion Middleware supported versions.

Refer to your component-specific installation and upgrade documentation for additional information about the schemas that are required for your upgrade.

- If you used an OID-based policy store in 11g, make sure that you create a new 12c (12.2.1.1) OPSS schema before you perform the upgrade. After the upgrade, the OPSS schema will still remain LDAP-based store.
- You can only upgrade schemas for products that are available for upgrade in the Oracle Fusion Middleware 12c (12.2.1.1) release. **Do not attempt to upgrade a domain that includes components that are not yet available for upgrade to 12c (12.2.1.1).**

3.9 Upgrading the Schemas with the Upgrade Assistant

Run the Upgrade Assistant to upgrade all the schemas in the existing domain to 12.2.1.1. Oracle recommends that you run the Upgrade Assistant as a non-SYSDBA user.

To upgrade the schemas:

1. Run the Upgrade Assistant from the 12.2.1.1 Oracle home by entering the following command:

On UNIX operating system:

```
$Oracle_Home/oracle_common/upgrade/bin/ua
```

On Windows operating system:

```
%Oracle_Home%\oracle_common\upgrade\bin\ua.bat
```

2. The Welcome screen provides an overview of the Upgrade Assistant and some information about important pre-upgrade tasks. Click **Next**.

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

3. On the All Schemas screen, select **All Schemas Used by a Domain** and click **Next**.

When you select **All Schemas Used by a Domain**, the Upgrade Assistant discovers and selects all components that have a schema available to upgrade. Additionally, the Upgrade Assistant pre-populates connection information on the schema input screens.

4. On the Component List screen, verify that all the components and schemas you want to upgrade within a domain are listed and click **Next**.

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

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.

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

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

6. On each Schema Credentials screen, specify the information required to connect to the selected schema and the database that hosts the schema on this screen.

The screen name changes based on the type of schema selected. For example, "MDS Schema".

Since the component ID or schema name is changed for UCSUMS schema as of release 12.1.2, 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.

Tip: The format of the database connect string is displayed below the text box. Specify the database connect string in the suggested format.

7. The Examine screen displays the status of the Upgrade Assistant as it examines each component, verifying that the component is ready for upgrade. If the status is "Examine finished.", click **Upgrade**.

If the examine phase fails, Oracle recommends that you cancel the upgrade by clicking **No** on the Examination Failure dialog box. Click **View Log** to see what caused the error and refer to Troubleshooting Your Upgrade 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** on 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.
-
-

8. On the Upgrade Summary screen, review the summary of the options you have selected by expanding the tree.

Review the Source Version and the Target Version to make sure that both the versions are correct before proceeding with the upgrade.

The response file collects and stores all the information that you have entered through the Upgrade Assistant's graphical user interface, and enables you to perform a silent upgrade at a later time. The silent upgrade performs exactly the same function that the Upgrade Assistant wizard 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** and provide the location and name of the response file.

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

9. The Upgrade Progress screen shows the status of the upgrade process and the projected Target Version of the component after a successful upgrade. Click **Next**.

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.

10. If the upgrade is successful, you see 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 the component functional in the new installation. This is an optional window that only appears if a component has post-upgrade steps.

11. If the upgrade has failed, you see the Upgrade Failure screen. This indicates that the upgrade of one or more components has failed. The components cannot be upgraded this time.

Click **View Log** to view and troubleshoot the errors.

Fix the issues in the pre-upgrade environment before starting the Upgrade Assistant again. Restore your pre-upgrade environment from backup (making sure to keep the original backup files in a separate location), fix the issues, and restart the Upgrade Assistant.

3.10 Reconfiguring the Domain with the Reconfiguration Wizard

The Reconfiguration Wizard reconfigures the domain while retaining the location of the domain. Use the Reconfiguration Wizard to upgrade your domain to the latest version.

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. Run the Reconfiguration Wizard by entering the following command:

On UNIX operating system:

```
$ORACLE_HOME/oracle_common/common/bin/reconfig.sh
```

On Windows operating system:

```
%ORACLE_HOME%\oracle_common\common\bin\reconfig.cmd
```

Note: When you run the `reconfig.cmd` or `reconfig.sh` command, you can get the following error message 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 including the `-Dpython.cachedir=valid_directory` parameter in the command.

Oracle recommends that you specify the “log” option while starting the Reconfiguration Wizard as shown in the following example:

```
./reconfig.sh -log=/$ORACLE_HOME/logs/reconfig.log -log_priority=ALL
```

You can set the `log_priority` as per your requirements.

4. 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**.
5. The Reconfiguration Setup Progress screen shows 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.
6. 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. 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.

7. 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.

You can configure the JDBC data sources defined in your domain source on the JDBC Data Sources screen. The JDBC data sources associated with the products for which you are creating the domain are listed in the lower half of the screen.

Select the data source(s) from the **Data Source Name** drop-down list 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, you get a pop-up with the following warning:

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.

8. 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: In order 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.

9. On the Database Configuration Type screen, select **RCU Data**.

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.

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

10. On the JDBC Component Schema Test screen, select all the component schemas and click **Test Selected Connections** to test the connection for each schema.

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

11. 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 3-6 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.
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	<p>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.</p> <p>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.</p> <p>For more information about Node Manager configuration, see <i>Administering Node Manager for Oracle WebLogic Server</i>.</p>
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 3-6 (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 nodemanager.properties file. Deselect this check box if you want to continue using the settings in the nodemanager.properties 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.

- 12.** 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**.

- 13.** 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.

- 14.** 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**.

15. 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 to indicate 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.

3.11 Upgrading the Domain Component Configurations with the Upgrade Assistant

Follow the instructions in this section to upgrade any additional domain component configurations, such as OWSM policy metadata structure and adapter configurations, using the Upgrade Assistant.

To upgrade the domain configurations:

1. Run the Upgrade Assistant from the 12.2.1.1 Oracle home by entering the following command:

On UNIX operating system:

```
$Oracle_Home/oracle_common/upgrade/bin/ua
```

On Windows operating system:

```
%Oracle_Home%\oracle_common\upgrade\bin\ua.bat
```

2. The Welcome screen provides an overview of the Upgrade Assistant and some information about important pre-upgrade tasks. Click **Next**.

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

3. 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**.
4. On the Component List screen, verify that all the components you want to upgrade within a domain are listed 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.

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

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

6. On the UMS Configuration screen, specify the login credentials of the remote managed servers hosting your UMS 11g configuration files. The Upgrade Assistant automatically copies remote configuration files if all necessary prerequisites are met and the required login information is provided.

Note: If the Upgrade Assistant is unable to locate the managed servers or the configuration files, you will have to manually copy the files and then restart the Upgrade Assistant. For more information, see [Error while Copying User Messaging Service \(UMS\) Configuration Files](#).

7. The Examine screen displays the status of the Upgrade Assistant as it examines each component, verifying that the component is ready for upgrade. If the status is "Examine finished.", click **Upgrade**.

If the examine phase fails, Oracle recommends that you cancel the upgrade by clicking **No** on the Examination Failure dialog box. Click **View Log** to see what caused the error and refer to *Troubleshooting Your Upgrade* 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** on 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.
-

8. On the Upgrade Summary screen, review the summary of the options you have selected by expanding the tree.

Review the Source Version and the Target Version to make sure that both the versions are correct before proceeding with the upgrade.

The response file collects and stores all the information that you have entered through the Upgrade Assistant's graphical user interface, and enables you to perform a silent upgrade at a later time. The silent upgrade performs exactly the same function that the Upgrade Assistant wizard 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** and provide the location and name of the response file.

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

9. The Upgrade Progress screen shows the status of the upgrade process and the projected Target Version of the component after a successful upgrade. Click **Next**.

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.

10. If the upgrade is successful, you see 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 the component functional in the new installation. This is an optional window that only appears if a component has post-upgrade steps.

11. If the upgrade has failed, you see the Upgrade Failure screen. This indicates that the upgrade of one or more components has failed. The components cannot be upgraded this time.

Click **View Log** to view and troubleshoot the errors.

Fix the issues in the pre-upgrade environment before starting the Upgrade Assistant again. Restore your pre-upgrade environment from backup (making sure to keep the original backup files in a separate location), fix the issues, and restart the Upgrade Assistant.

3.12 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 errors during the Upgrade phase. The environment should be considered unstable and will need to be restored to its pre-upgrade state.

For more information, 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.

3.12.1 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.

3.12.2 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 *Oracle Fusion Middleware 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 (businessterms.xml). These files are located in the /applications folder for each managed server, as shown in [Table 3-7](#).

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

procedures from [Upgrading the Domain Component Configurations with the Upgrade Assistant](#)

Table 3-7 Configuration File Locations

Configuration file	Location
UMS Server (appconfig.xml)	\$DOMAIN_HOME/config/fmwconfig/ servers/MANAGED_SERVER_NAME/ applications/usermessagingserver/ configuration/appconfig.xml
Driver Configuration (driverconfig.xml)	\$DOMAIN_HOME/config/fmwconfig/ servers/MANAGED_SERVER_NAME/ applications/usermessagingdriver- DRIVER_NAME/configuration/ driverconfig.xml
User Preferences (businessterms.xml)	\$DOMAIN_HOME/config/fmwconfig/ servers/MANAGED_SERVER_NAME/ 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.

3.13 Performing the Post-Upgrade Tasks

After you upgrade Oracle Fusion Middleware 11g Application Developer to Oracle Fusion Middleware 12c Infrastructure, you must complete the post-upgrade tasks.

For post-upgrade tasks, see [Tasks to Perform After Upgrade](#).

Tasks to Perform After Upgrade

The topics in this section describe the tasks you might have to perform after upgrading to Oracle Fusion Middleware 12c Infrastructure. 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 the Enterprise Manager.

[Starting and Stopping Servers in the Correct Order](#)

Oracle recommends you to start and stop the servers in a particular order to avoid issues with the deployment.

[Verifying the Domain-specific-Component Configurations Upgrade](#)

To verify that the domain-specific-component configurations upgrade was successful, log in to the Administration console and the Fusion Middleware Control and verify that the version numbers for each component is 12.2.1.

[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.

4.1 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 the Enterprise Manager.

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.

For more information, see [Starting and Stopping Servers in the Correct Order](#).

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`

4.2 Starting and Stopping Servers in the Correct Order

Oracle recommends you to start and stop the servers in a particular order to avoid issues with the deployment.

After the Infrastructure upgrade, start all of the Administration and Managed servers for your environment and make sure that they are functioning as expected.

Note:

Procedures for starting and stopping Oracle Fusion Middleware, including the Administration Server, Managed Servers, and components are provided in Starting and Stopping Oracle Fusion Middleware in *Administering Oracle Fusion Middleware*.

Start servers in the following order:

1. Node Managers
2. Administration Server
3. Webtier (including the Oracle HTTP Server)
4. Oracle Web Services Manager (OWSM) Managed Server (if installed)
5. Service-Oriented Architecture (SOA) Managed Server (if installed)
6. Oracle Service Bus (OSB) Managed Server (if installed)
7. Business Activity Monitoring (BAM) Managed Server (if installed)

Stop servers in the following order:

1. Business Activity Monitoring (BAM) Managed Server (if installed)
2. Oracle Service Bus (OSB) Managed Server (if installed)
3. Service-Oriented Architecture (SOA) Managed Server (if installed)
4. Oracle Web Services Manager (OWSM) Managed Server (if installed)
5. Webtier (including the Oracle HTTP Server)
6. Admin Server
7. Node Managers

For more information on stopping servers, see Starting and Stopping Oracle Fusion Middleware in *Administering Oracle Fusion Middleware*.

Starting the Node Manager

Node Manager is a Java utility that runs as separate process from WebLogic Server and allows you to perform common operations for a

Managed Server, regardless of its location with respect to its Administration Server.

Starting the Administration Server

The Administration Server provides a central point for managing a WebLogic Server domain. All other WebLogic Server instances in a domain are called Managed Servers. In a domain with only a single WebLogic Server instance, that server functions both as Administration Server and Managed Server.

Starting the Webtier (Oracle HTTP Server)

Oracle HTTP Server is the Web server component for Oracle Fusion Middleware. It provides a listener for Oracle WebLogic Server and the framework for hosting static pages, dynamic pages, and applications over the Web.

4.2.1 Starting the Node Manager

Node Manager is a Java utility that runs as separate process from WebLogic Server and allows you to perform common operations for a Managed Server, regardless of its location with respect to its Administration Server.

Note:

To use a per-host Node Manager configuration, ensure that –
`Dohs.product.home=<MW_HOME>` is set for `JAVA_OPTIONS` in the appropriate Node Manager script. Run the command `NodeManager.cmd | sh` if you are not using the Node Manager service. Run the command `installNodeMgrSvc.cmd` if you are using the Node Manager service.

The Host and Port used should match those used with your Upgraded setup. For more information on how to edit the Host/Port values in the file to install the Node Manager Service with the correct details, see [Configuring Java Node Manager](#) in *Oracle Fusion Middleware Node Manager Administrator's Guide for Oracle WebLogic Server*

To start the Node Manager:

1. Change directory to `DOMAIN_HOME/bin`
2. On the Unix operating system, enter the following command: `nohup ./startNodeManager.sh > nm.out&`

Where `nohup` and `nm.out` are sample output files

On the Windows operating systems, enter the following command:
`startNodeManager.cmd`

4.2.2 Starting the Administration Server

The Administration Server provides a central point for managing a WebLogic Server domain. All other WebLogic Server instances in a domain are called Managed Servers. In a domain with only a single WebLogic Server instance, that server functions both as Administration Server and Managed Server.

To start an Administration Server:

1. On Unix operating system, change directory to `DOMAIN_HOME/bin`

On the Windows operating system, change directory to `DOMAIN_HOME\bin`

2. On the Unix operating system, enter the following command: `./startWebLogic.sh`

On the Windows operating system, enter the following command:
`startWebLogic.cmd`

To verify that your domain is reconfigured successfully, log in to the Administration console using the following URL, and verify that the version number displayed on the console is 12.2.1:

`http://administration_server_host:administration_server_port/console`

4.2.3 Starting the Webtier (Oracle HTTP Server)

Oracle HTTP Server is the Web server component for Oracle Fusion Middleware. It provides a listener for Oracle WebLogic Server and the framework for hosting static pages, dynamic pages, and applications over the Web.

To start Oracle HTTP Server:

1. Change directory to: `DOMAIN_HOME/bin`.
2. On the Unix operating system, enter the following command:
`startComponent.sh ohs_name`

On the Windows operating system, enter the following command:
`startComponent.cmd ohs_name`

For more information, see Starting and Stopping System Components in *Administering Oracle Fusion Middleware*.

4.3 Verifying the Domain-specific-Component Configurations Upgrade

To verify that the domain-specific-component configurations upgrade was successful, log in to the Administration console and the Fusion Middleware Control and verify that the version numbers for each component is 12.2.1.

To log into the Administration Console, go to: `http://administration_server_host:administration_server_port/console`

To log into the Fusion Middleware Control, go to: `http://administration_server_host:administration_server_port/em`

Note:

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

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**.

4.4 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.

For more information, 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 \(Optional\)](#).

4.5 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 [Configuring and Managing Auditing](#) in *Fusion Middleware Application Security Guide*.

4.6 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`. For more information, 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. For more information, see [Disabling a Globally Attached Policy in *Securing WebLogic Web Services for Oracle WebLogic Server*](#)

4.7 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 4-1](#) lists some common administration tasks you will likely want to perform after upgrading to Infrastructure 12c.

Table 4-1 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> .
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> .

4.8 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](#)

- [About Oracle 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.

4.8.1 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:

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

4.8.2 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.

4.8.2.1 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 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*.

4.8.2.2 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.

For more information, see Migrating to Oracle JDeveloper 12.2.1 From a Previous Version in *Installing Oracle JDeveloper*.

4.8.2.3 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 Developing Asynchronous Web Services in *Developing Oracle Infrastructure Web Services*.

4.9 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/
sampledomaintemplate.jar -app_dir=$DOMAIN_HOME/user_projects/
applications -overwrite_domain=true
```

Starting the Upgrade Assistant

This topic describes how to run the Oracle Fusion Middleware Upgrade Assistant. You can run the Upgrade Assistant either in the Graphical User Interface (GUI) mode or in the response file mode.

You can use the optional command-line interface arguments, documented in this topic, to upgrade your Oracle Fusion Middleware components to the latest version. In addition, a response file can be generated to automate some of the upgrade tasks.

This topic contains the following sections:

[Starting the Upgrade Assistant in Graphical User Interface \(GUI\) Mode](#)

The Upgrade Assistant is used to upgrade schemas, component configurations and standalone system components.

[Starting the Upgrade Assistant in Response File Mode](#)

Silent or “hands free” upgrades can be performed using a response file. The response file can only be created after you have provided the information in the Upgrade Assistant screens.

[Setting the DISPLAY Environment Variable](#)

A.1 Starting the Upgrade Assistant in Graphical User Interface (GUI) Mode

The Upgrade Assistant is used to upgrade schemas, component configurations and standalone system components.

Oracle recommends that you successfully complete the upgrade of schemas and component configurations for a single domain before beginning the upgrade of another domain.

Note: The Upgrade Assistant should be run by a non-SYSDBA user whenever possible. The steps to create a user who has the privileges required to upgrade the schemas are described in [Creating a Non-SYSDBA User](#).

To start the Upgrade Assistant:

1. On UNIX operating systems:: change directory to `ORACLE_HOME/oracle_common/upgrade/bin`.

On Windows operating systems: change directory to `ORACLE_HOME\oracle_common\upgrade\bin`.

2. Enter the following command to start the Upgrade Assistant:

On UNIX operating systems:

```
./ua
```

On Windows operating systems:

```
ua.bat
```

You can also launch the Upgrade Assistant with logging parameters as shown in the UNIX example below:

```
./ua [-logLevel <log_level>] [-logDir <log_directory>]
```

Logging level. Select one of the following:

- TRACE
- NOTIFICATION
- WARNING
- ERROR
- INCIDENT_ERROR

The default logging level is NOTIFICATION.

Note: When troubleshooting, consider setting the `-logLevel` to TRACE so that more information will be logged. If additional information is not needed, change the `logLevel` as the Upgrade Assistant's log files can become very large when `-logLevel TRACE` is used.

Starting the Upgrade Assistant with Additional Parameters (Optional)

A.1.1 Starting the Upgrade Assistant with Additional Parameters (Optional)

[Table A-1](#) lists the command-line parameters you can use while running the Upgrade Assistant in the GUI mode. The following example shows the usage of these parameters on your respective operating system:

On UNIX operating systems:

Change directory to `ORACLE_HOME/oracle_common/upgrade/bin`

Enter the command: `./ua -help`

On Windows operating systems:

Change directory to `ORACLE_HOME\oracle_common\upgrade\bin`

Enter the command: `ua.bat -help`

Note: If you get an Xlib error when starting the Oracle Upgrade Assistant such as "Failed to connect to server", "Connection refused by server", or "Can't open display", then you must set the DISPLAY environment variable and restart the Upgrade Assistant as described in [Setting the DISPLAY Environment Variable](#).

Table A-1 Upgrade Assistant GUI Command Line Parameters

Table A-1 (Cont.) Upgrade Assistant GUI Command Line Parameters

Parameter	Required or Optional Parameter?	Description
-logLevel	Optional.	<p>Logging level. Select one of the following:</p> <ul style="list-style-type: none"> • TRACE • NOTIFICATION • WARNING • ERROR • INCIDENT_ERROR <p>The default logging level is NOTIFICATION.</p>

Note:

When troubles hooting, consider setting the -logLevel to TRACE so that more information will be logged. If additional information is not needed, change the logLevel as the Upgrade Assistant's log files can become very large when -logLevel is TRACE is used.

Table A-1 (Cont.) Upgrade Assistant GUI Command Line Parameters

Parameter	Required or Optional Parameter?	Description
-logDir	Optional.	<p>Change the default location of upgrade log files and temporary files. You must specify an existing, writable directory where Upgrade Assistant will create log files and temporary files.</p> <p>On UNIX operating systems, the default locations are:</p> <p><i>ORACLE_HOME</i>/oracle_common/upgrade/logs <i>ORACLE_HOME</i>/oracle_common/upgrade/temp</p> <p>On Windows operating systems, the default locations are:</p> <p><i>ORACLE_HOME</i>\oracle_common\upgrade\logs <i>ORACLE_HOME</i>\oracle_common\upgrade\temp</p> <p>Note: If you have installed the Upgrade Assistant in a read-only directory, you must specify this parameter to ensure the log directory is writable. If the log directory is read-only, the Upgrade Assistant will fail.</p>
-threads	Optional	<p>Identify the number of threads available for concurrent schema upgrades or readiness checks of the schemas.</p> <p>The value must be a positive integer in the range 1 to 8. Default is 4 threads.</p>

Table A-1 (Cont.) Upgrade Assistant GUI Command Line Parameters

Parameter	Required or Optional Parameter?	Description
-readiness	Required for Readiness Check	Performs the upgrade readiness check without performing any actual examines or upgrades. Schemas and configuration are checked.
		<p style="text-align: right;">Note: -readiness option may not appear on the command line with -examine option.</p> <p style="text-align: right;">For more information, see Performing the Readiness Check</p>
-help	Optional.	Prints all the command-line options to the console.

A.2 Starting the Upgrade Assistant in Response File Mode

Silent or “hands free” upgrades can be performed using a response file. The response file can only be created after you have provided the information in the Upgrade Assistant screens.

The following topics describe how you can upgrade the supported Oracle Fusion Middleware components using a response file. The response file collects all the information that you have entered through the Upgrade Assistant's graphical user interface screens, and performs exactly the same function that the Upgrade Assistant wizard performs.

Note: You must run the Upgrade Assistant in GUI mode first to generate the response file that will be used to complete your silent upgrade.

[Creating an Upgrade Response File](#)

[Using the Response File to Upgrade Fusion Middleware](#)

A.2.1 Creating an Upgrade Response File

The **Save Response File** option on the Upgrade Summary screen creates a file that uses the information you have already provided in the Upgrade Assistant screens. The

response file enables you to use the saved information instead of manually entering data through the Upgrade Assistant wizard screens.

Once you select the **Save Response File** option, you will be prompted for a name and location where you want to create this response file. After it is created, you can use it exactly as-is to replicate the upgrade options on other systems, or modify it as needed.

For more information, see [Using the Response File to Upgrade Fusion Middleware](#).

A.2.2 Using the Response File to Upgrade Fusion Middleware

To perform upgrades using a response file from the command-line interface (CLI), use the following command:

On Unix operating systems:

Change directory to `ORACLE_HOME/oracle_common/upgrade/bin`

Execute the following:

```
./ua -response <response_file> [-examine] [-logLevel
<log_level>] [-logDir <log_directory>] [-threads <number>]
```

On Windows operating systems:

Change directory to `ORACLE_HOME\oracle_common\upgrade\bin`

Execute the following:

```
ua.bat -response <response_file> [-examine] [-logLevel
<log_level>] [-logDir <log_directory>] [-threads <number>]
```

Table A-2 Upgrade Assistant Response File Mode Command Line Parameters

Parameter	Required or Optional Parameter?	Description
<code>-readiness</code>	Required for Readiness Check	Performs the upgrade readiness check without performing any actual examines or upgrades. Schemas and configuration are checked.

Note:
Do not specify this parameter if you have specified the `-examine` parameter.

For more information, see [Performing the Readiness Check](#)

Table A-2 (Cont.) Upgrade Assistant Response File Mode Command Line Parameters

Parameter	Required or Optional Parameter?	Description
-threads	Optional	Identify 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. Default is 4 threads.
-response	Required.	File containing inputs required to perform an upgrade. This file can be generated from inputs entered when the Upgrade Assistant is run in graphical mode.
-examine	Optional.	If this option is present, Upgrade Assistant performs the examine phase but DOES NOT perform any actual upgrades.

Note:

Do not specify this parameter if you have specified the -readiness parameter.

Table A-2 (Cont.) Upgrade Assistant Response File Mode Command Line Parameters

Parameter	Required or Optional Parameter?	Description
-logLevel	Optional.	<p>Logging level. Select one of the following:</p> <ul style="list-style-type: none"> • TRACE • NOTIFICATION • WARNING • ERROR • INCIDENT_ERROR <p>The default logging level is NOTIFICATION.</p>

Note:
Consider setting the -logLevel to TRACE so that more information will be logged. This will be useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if -logLevel TRACE is used.

Table A-2 (Cont.) Upgrade Assistant Response File Mode Command Line Parameters

Parameter	Required or Optional Parameter?	Description
-logDir	Optional.	<p>Change the default location of upgrade log files and temporary files. You must specify an existing, writable directory where Upgrade Assistant will create log files and temporary files.</p> <p>On UNIX operating systems, the default locations are:</p> <pre>ORACLE_HOME/ oracle_common/upgrade/ logs ORACLE_HOME/ oracle_common/upgrade/ temp</pre> <p>On Windows operating systems, the default locations are:</p> <pre>ORACLE_HOME \oracle_common\upgrade \logs ORACLE_HOME \oracle_common\upgrade \temp</pre>
-help	Optional.	View all of the command line options.

A.3 Setting the DISPLAY Environment Variable

When running Upgrade Assistant in GUI mode, you must set the DISPLAY variable properly or you may receive errors such as:

Xlib: connection to ":1.0" refused by server

Xlib: No protocol specified

Cause: These errors indicate that the DISPLAY variable is not set up properly to allow a GUI to be displayed to the screen.

Action: Set the DISPLAY environment variable to the system name or IP address of your local workstation, and re-run Upgrade Assistant.

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

