

Oracle® Fusion Middleware

Upgrading Oracle HTTP Server

12c (12.2.1.1)

E71495-01

June 2016

This guide describes an upgrade of the Oracle HTTP Server to this release of Oracle Fusion Middleware.

Oracle Fusion Middleware Upgrading Oracle HTTP Server , 12c (12.2.1.1)

E71495-01

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

Primary Author: Priyanka Chheda

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.....	ix
Other Significant Changes in this Document for 12c	ix
1 Understanding the Oracle HTTP Server Upgrade to 12c	
1.1 Differences between Oracle HTTP Server 11g and 12c	1-1
1.2 Fusion Middleware Infrastructure Upgrade Topology with Oracle HTTP Server	1-2
1.3 Determining whether Oracle HTTP Server is Standalone or Managed (Collocated)	1-4
2 Preparing to Upgrade Oracle HTTP Server	
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

Part I Upgrading a Standalone Oracle HTTP Server

3 Introduction to the Standalone Oracle HTTP Server Upgrade

3.1	Understanding the Standalone Oracle HTTP Server Topology.....	3-1
3.2	Overview of the Standalone Oracle HTTP Server Upgrade Process	3-3
3.2.1	Flow Chart of the Standalone Upgrade Process from 11g to 12c	3-3
3.2.2	Roadmap for Upgrading a Standalone Oracle HTTP Server from 11g to 12c.....	3-4
3.2.3	Flow Chart of the Standalone Oracle HTTP Server Upgrade Process from a Previous 12c Release.....	3-6
3.2.4	Roadmap for Upgrading a Standalone Oracle HTTP Server from a Previous 12c Release	3-6

4 Upgrading a Standalone Oracle HTTP Server from 11g to 12c

4.1	Important Pre-Upgrade Considerations.....	4-1
4.1.1	Oracle Web Cache 11g Users	4-2
4.1.2	WebGate 11g Users	4-2
4.1.3	Updating the WebGate log file-name.....	4-3
4.1.4	Application Artifacts from 11g.....	4-3
4.1.5	Migrating 11g Application Artifacts.....	4-3
4.2	Upgrading an 11g Standalone Oracle HTTP Server	4-4
4.2.1	Installing the Standalone Oracle HTTP Server	4-4
4.2.2	Upgrading the 11g Domain using Upgrade Assistant.....	4-6
4.2.3	Verifying the Upgrade.....	4-9

5 Upgrading a Standalone Oracle HTTP Server from a Previous 12c Release

5.1	Installing the Standalone Oracle HTTP Server.....	5-1
5.2	Reconfiguring the Existing Domain with the Reconfiguration Wizard	5-3
5.3	Upgrading the 12c Domain using Upgrade Assistant	5-8
5.4	Verifying the Upgrade	5-10
5.4.1	Starting the Node Manager.....	5-11
5.4.2	Starting the Standalone Oracle HTTP Server	5-11

Part II Upgrading a Managed Oracle HTTP Server

6 Introduction to the Managed Oracle HTTP Server Upgrade

6.1	Flowchart of Managed Oracle HTTP Server Upgrade Process.....	6-1
6.2	Roadmap for Upgrading a Managed Oracle HTTP Server	6-2

7 Upgrading a Managed Oracle HTTP Server from 11g to 12c

7.1	Installing the Managed 12c (12.2.1.1.0) Oracle HTTP Server	7-1
7.2	Creating the Required Schemas before Upgrade.....	7-3
7.3	Reconfiguring the 11g Domain with the Reconfiguration Wizard	7-6

7.4	Upgrading the Component Configuration with the Upgrade Assistant.....	7-11
7.5	Post-Upgrade Tasks.....	7-13
7.5.1	Starting the Node Manager.....	7-13
7.5.2	Starting the Administration Server.....	7-14
7.5.3	Starting the Oracle HTTP Server.....	7-15
7.5.4	Verifying that Oracle HTTP Server Installation is Successful	7-16
8	Upgrading a Managed Oracle HTTP Server from a Previous 12c Release	
8.1	Installing the Managed 12c (12.2.1.1.0) Oracle HTTP Server	8-1
8.2	Upgrading the Existing 12c Schema with the Upgrade Assistant.....	8-3
8.3	Reconfiguring the Existing Domain with the Reconfiguration Wizard	8-5
8.4	Upgrading the Component Configuration with the Upgrade Assistant.....	8-10
8.5	Post-Upgrade Tasks.....	8-12
8.5.1	Starting the Node Manager.....	8-13
8.5.2	Starting the Administration Server.....	8-13
8.5.3	Starting the Oracle HTTP Server.....	8-14
8.5.4	Verifying that Oracle HTTP Server Installation is Successful	8-15

Preface

This preface contains the following sections:

[Audience](#)

[Documentation Accessibility](#)

[Related Documents](#)

[Conventions](#)

Audience

This manual is intended for Oracle Fusion Middleware system administrators who are responsible for upgrading Oracle Fusion Middleware. It is assumed that the readers of this manual have knowledge of the following:

- Oracle Fusion Middleware 11g or Oracle Fusion Middleware 12c system administration and configuration information for the existing deployment
- The configuration and expected behavior of the system or systems being upgraded

Documentation Accessibility

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

Access to Oracle Support

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

Related Documents

For more information, see the following related documentation available in the Oracle Fusion Middleware 12c documentation library:

- *Planning an Upgrade of Oracle Fusion Middleware*
- *Installing and Configuring Oracle HTTP Server*
- *Administering Oracle HTTP Server*

- *Administering Oracle Fusion Middleware*
- *Understanding Oracle Fusion Middleware*
- *Upgrading to the Oracle Fusion Middleware Infrastructure*

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 following topics introduce the new and changed features of the standalone Oracle HTTP Server and provides pointers to additional information.

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

[Other Significant Changes in this Document for 12c](#)

New and Changed Features for 12c

Before you begin the upgrade process, review the following list of changes for standalone Oracle HTTP Servers in 12c:

- The installer will no longer create a separate instance directory for Oracle HTTP Server. Instance information for the standalone OHS will be stored in a new **standalone domain** home. For more information, see [Understanding the Standalone Oracle HTTP Server Topology](#).
- OPMN agent and `opmnctl` command line utility are no longer used in Oracle Fusion Middleware. Instead, system components are managed by NodeManager and WLST commands.

Other Significant Changes in this Document for 12c

None.

Understanding the Oracle HTTP Server Upgrade to 12c

Before you upgrade your Oracle HTTP Server to 12c, make sure that you understand the differences from your Oracle HTTP Server 11g deployment.

Differences between Oracle HTTP Server 11g and 12c

There are a few key differences to be aware of before you upgrade your Oracle HTTP Server to this release of Oracle Fusion Middleware.

Fusion Middleware Infrastructure Upgrade Topology with Oracle HTTP Server

This topic contains the Oracle Fusion Middleware 11g Application Developer standard upgrade topology with Oracle HTTP Server and the resulting Oracle Fusion Middleware 12c Infrastructure topology as it appears after you complete the upgrade procedures in this guide.

Determining whether Oracle HTTP Server is Standalone or Managed (Collocated)

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. If you configure Oracle HTTP Server in a WebLogic Server domain, it is called as the Managed Oracle HTTP Server because you can manage the Oracle HTTP Server instances like any other elements of the WebLogic Server domain using Enterprise Manager Fusion Middleware Control, or WLST Command line interface, or the WebLogic Server Node Manager. If you install the Oracle HTTP software in a separate Oracle home without installing the Oracle Fusion Middleware Infrastructure, it is called as the standalone mode.

1.1 Differences between Oracle HTTP Server 11g and 12c

There are a few key differences to be aware of before you upgrade your Oracle HTTP Server to this release of Oracle Fusion Middleware.

[Table 1-1](#) lists the key differences between Oracle HTTP Server 11g and 12c.

Table 1-1 Differences between Oracle HTTP Server 11g and 12c

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

Table 1-1 (Cont.) Differences between Oracle HTTP Server 11g and 12c

In Oracle HTTP Server 11g:	In Oracle HTTP Server 12c:
Oracle HTTP Server instances are managed using the Oracle Process Manager and Notification Server (OPMN) management software. Optionally, the Oracle HTTP Server instances can be associated with the WebLogic domain.	When configured as part of an Oracle Application Server 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 Oracle HTTP Server instances.
mod_osso is supported and included with Oracle HTTP Server 11g. If you use mod_osso in 11g, it will be disabled after upgrade as it is not supported in 12c.	mod_osso is not supported or included with Oracle HTTP Server 12c. Oracle WebGate is the recommended replacement. You can install WebGate with Oracle HTTP Server. For more information on configuring WebGate with Oracle HTTP Server, see <i>Oracle Fusion Middleware Installing and Configuring Oracle HTTP Server</i> .

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 the following:

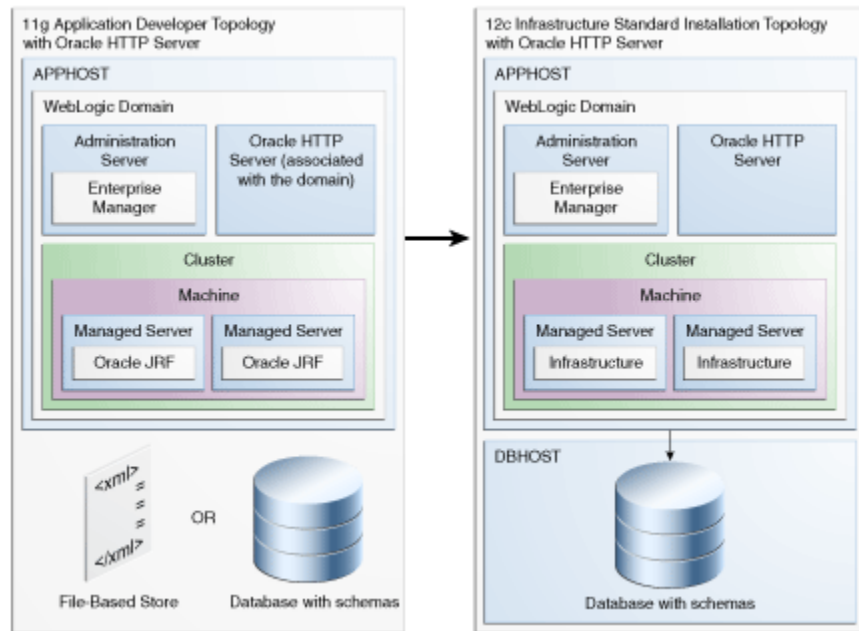
- New and Changed Features for 12c
- New and Deprecated Terminology for 12c
- What is the WebLogic Management Framework?

1.2 Fusion Middleware Infrastructure Upgrade Topology with Oracle HTTP Server

This topic contains the Oracle Fusion Middleware 11g Application Developer standard upgrade topology with Oracle HTTP Server 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 Standard Upgrade Topology with Oracle HTTP Server



Most of the elements in [Figure 1-1](#) are described in [Table 1-2](#).

Table 1-2 Description of the Elements in the Infrastructure Standard Upgrade Topology with Oracle HTTP Server

Element	Description and Links to Additional Documentation
11g Application Developer Topology with Oracle HTTP Server	<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>Figure 1-1 also shows an Oracle HTTP Server instance as part of the 11g domain.</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 with Oracle HTTP Server	<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> <p>Figure 1-1 also shows an Oracle HTTP Server instance as part of the 12c domain.</p>

Table 1-2 (Cont.) Description of the Elements in the Infrastructure Standard Upgrade Topology with Oracle HTTP Server

Element	Description and Links to Additional Documentation
Oracle HTTP Server "associated with the domain"	An Oracle HTTP Server 11g instance that has been configured to be "associated with" the Oracle WebLogic Server domain. In Oracle Fusion Middleware 11g, system component instance, such as Oracle HTTP Server, are configured with an Oracle Universal Installer-based configuration wizard and are managed using Oracle Process Manager and Notification Server.
Oracle HTTP Server	Unlike the Oracle HTTP Server 11g instance in the left side of the diagram, the Oracle HTTP Server 12c instance shown in the 12c topology is configured as part of the domain using the Oracle Fusion Middleware Configuration Wizard. It is managed using Oracle Enterprise Manager Fusion Middleware Control, the Oracle WebLogic Scripting Tool (WLST), and the Oracle WebLogic Server Node Manager software.

1.3 Determining whether Oracle HTTP Server is Standalone or Managed (Collocated)

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. If you configure Oracle HTTP Server in a WebLogic Server domain, it is called as the Managed Oracle HTTP Server because you can manage the Oracle HTTP Server instances like any other elements of the WebLogic Server domain using Enterprise Manager Fusion Middleware Control, or WLST Command line interface, or the WebLogic Server Node Manager. If you install the Oracle HTTP software in a separate Oracle home without installing the Oracle Fusion Middleware Infrastructure, it is called as the standalone mode.

To determine whether you are upgrading a managed or a standalone Oracle HTTP Server:

If you are an 11g user: Check for the registered property in the file `$ORACLE_INSTANCE/config/OPMN/opmn/instance.properties`. If it is set to true, then the instance is registered. A managed Oracle HTTP Server is registered, if it is not registered then it is Standalone Oracle HTTP Server.

If you are a 12c user: Check the element `extention-template-ref` and its attribute name in the file `$DOMAIN_HOME/init-info/domain-info.xml`. If you find an element with the name **Oracle HTTP Server (Standalone)**, then it is a standalone Oracle HTTP Server. And if you find an element with name **Oracle HTTP Server (Collocated)**, then it is collocated.

Preparing to Upgrade Oracle HTTP Server

Upgrade is performed while the servers are down. The pre-upgrade tasks 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) 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).</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) 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).

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).

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_$ASM_DISKGROUP to FMW with grant option;
grant select on v$xatrans$ to FMW with grant option;
grant execute on sys.dbms_system to FMW with grant option;
grant execute on DBMS_SCHEDULER to FMW with grant option;
grant select on dba_data_files to FMW with grant option;
grant execute on UTL_RAW to FMW with grant option;
grant execute on DBMS_XMLDOM to FMW with grant option;
grant execute on DBMS_APPLICATION_INFO to FMW with grant option;
grant execute on DBMS_UTILITY to FMW with grant option;
grant execute on DBMS_SESSION to FMW with grant option;
grant execute on DBMS_METADATA to FMW with grant option;
grant execute on DBMS_XMLGEN to FMW with grant option;
grant execute on DBMS_DATAPUMP to FMW with grant option;
grant execute on DBMS_MVIEW to FMW with grant option;
grant select on ALL_ENCRYPTED_COLUMNS to FMW with grant option;
grant select on dba_queue_subscribers to FMW with grant option;
grant execute on SYS.DBMS_ASSERT to FMW with grant option;
```

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.

Part I

Upgrading a Standalone Oracle HTTP Server

Part I contains the following chapters:

[Introduction to the Standalone Oracle HTTP Server Upgrade](#)

[Upgrading a Standalone Oracle HTTP Server from 11g to 12c](#)

[Upgrading a Standalone Oracle HTTP Server from a Previous 12c Release](#)

Introduction to the Standalone Oracle HTTP Server Upgrade

This chapter provides a general introduction to the Oracle Fusion Middleware Standalone Oracle HTTP Server and an overview of the steps you must perform to upgrade from an Oracle Fusion Middleware 11g or a previous 12c Release (12.1.2 and above) of Oracle HTTP Server to this Oracle HTTP Server 12c Release (12.2.1.1).

Caution:

This section should be used to upgrade an Oracle HTTP Server that is NOT managed through a WebLogic Server domain.

If your Oracle HTTP Server is managed through Oracle WebLogic, then follow the procedure in [Upgrading a Managed Oracle HTTP Server](#) .

This chapter includes the following sections:

[Upgrading a Standalone Oracle HTTP Server](#)

[Understanding the Standalone Oracle HTTP Server Topology](#)

In 12c, a **standalone Oracle HTTP Server** is not managed by or registered to an Oracle WebLogic Server domain. A standalone Oracle HTTP Server 12c topology is installed and configured without the Oracle Fusion Middleware Infrastructure. A **managed Oracle HTTP Server**, however, is associated with an existing Oracle WebLogic Server domain. For the standalone scenario, you install the Oracle HTTP Server software in its own Oracle home, and you configure the Oracle HTTP Server instance in its own **standalone domain**.

[Overview of the Standalone Oracle HTTP Server Upgrade Process](#)

The valid starting points for upgrading the Oracle HTTP Server are 11g Release 11.1.1.7 and above and 12c Release 12.1.2 and above.

3.1 Understanding the Standalone Oracle HTTP Server Topology

In 12c, a **standalone Oracle HTTP Server** is not managed by or registered to an Oracle WebLogic Server domain. A standalone Oracle HTTP Server 12c topology is installed and configured without the Oracle Fusion Middleware Infrastructure. A **managed Oracle HTTP Server**, however, is associated with an existing Oracle WebLogic Server domain. For the standalone scenario, you install the Oracle HTTP Server software in its own Oracle home, and you configure the Oracle HTTP Server instance in its own **standalone domain**.

Note:

- For more information on installing and configuring a standalone Oracle HTTP Server, see *Installing and Configuring Oracle HTTP Server*.
- For more information on the latest 12c standalone domain, see *What Is a Standalone Domain?* in *Understanding Oracle Fusion Middleware*.
- For more information on the administration scenarios and key features of the Oracle HTTP Server, see *Introduction to Oracle HTTP Server Administrator's Guide for Oracle HTTP Server*.
- For more information on upgrading a managed Oracle HTTP Server, see *Performing the Infrastructure Upgrade* in *Upgrading to the Oracle Fusion Middleware Infrastructure*.

Figure 3-1 shows a standalone Oracle HTTP Server topology.

Figure 3-1 Standalone Oracle HTTP Server Upgrade Topology

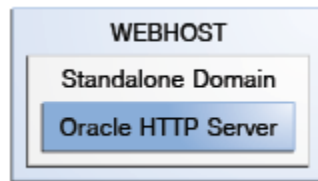


Table 3-1 describe the elements of this topology

Table 3-1 Description of the Elements in the Oracle Fusion Middleware Standalone Oracle HTTP Server Upgrade Topology

Element	Description and Links to Additional Documentation
WEBHOST	Standard term used in Oracle documentation referring to the machine that hosts the Web tier.
Standalone Domain	<p>A standalone domain is only created if you are upgrading from 11g to 12c. If you are upgrading from 12.1.2 or 12.1.3, then the standalone domain already exists and there is no need to create the standalone domain.</p> <p>The standalone domain has a directory structure similar to an Oracle WebLogic domain, but it does not contain an Administration Server or Managed Servers. The Oracle WebLogic Server Node Manager and other tools allow you to manage the standalone Oracle HTTP Server instance.</p> <p>For more information on standalone domains, see <i>What Is a Standalone Domain?</i> in <i>Understanding Oracle Fusion Middleware</i>.</p>

3.2 Overview of the Standalone Oracle HTTP Server Upgrade Process

The valid starting points for upgrading the Oracle HTTP Server are 11g Release 11.1.1.7 and above and 12c Release 12.1.2 and above.

The following sections provide a high level overview of the procedures you need to perform to upgrade your standalone Oracle HTTP Server:

[Flow Chart of the Standalone Upgrade Process from 11g to 12c](#)

[Roadmap for Upgrading a Standalone Oracle HTTP Server from 11g to 12c](#)

[Flow Chart of the Standalone Oracle HTTP Server Upgrade Process from a Previous 12c Release](#)

[Roadmap for Upgrading a Standalone Oracle HTTP Server from a Previous 12c Release](#)

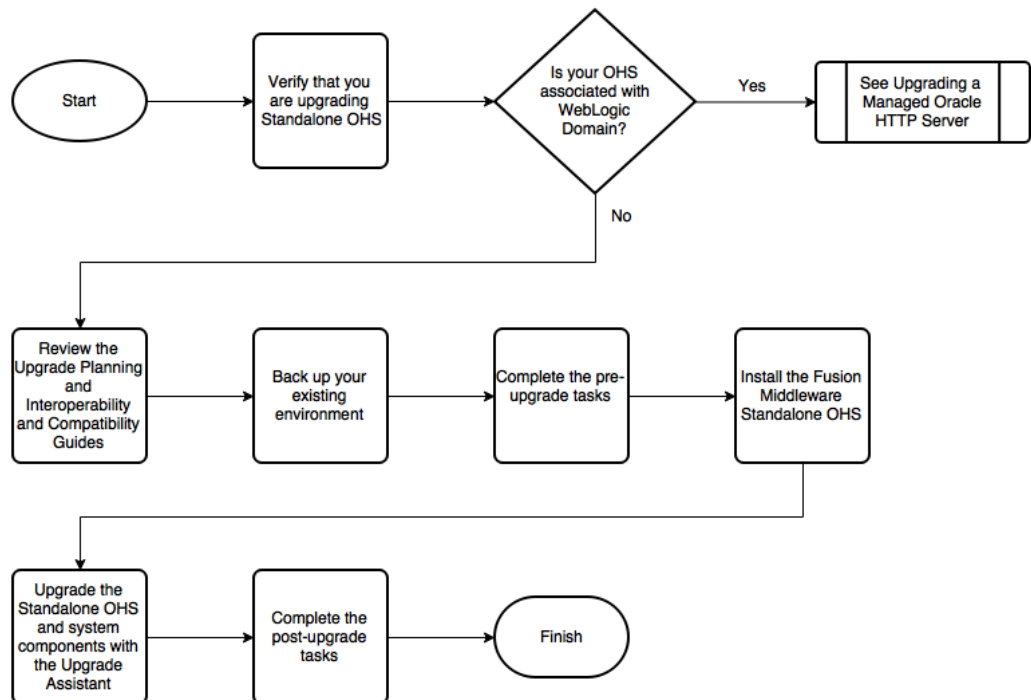
3.2.1 Flow Chart of the Standalone Upgrade Process from 11g to 12c

Figure 3-2 shows the high-level procedures associated with a standalone Oracle HTTP Server upgrade when the starting point is 11g. The tools used for each step are also listed.

Detailed instructions for upgrading your standalone Oracle HTTP Server are described in [Upgrading a Standalone Oracle HTTP Server from 11g to 12c](#) .

Note: Before starting the upgrade, you must shutdown the 11g instance.

Figure 3-2 Standalone Oracle HTTP Server Upgrade from 11g to 12c



3.2.2 Roadmap for Upgrading a Standalone Oracle HTTP Server from 11g to 12c

[Table 3-2](#) describes the tasks that must be completed to upgrade a standalone Oracle HTTP Server from 11g to 12c.

Table 3-2 Roadmap for Standalone Oracle HTTP Server Upgrade Process

Step No.	Importance (Mandatory, Conditional, or Optional)	Task	Description
1	Recommended.	Verify that you are upgrading a standalone Oracle HTTP Server.	To determine which Oracle HTTP Server you have in your existing environment, see Determining whether Oracle HTTP Server is Standalone or Managed (Collocated) .
2	Optional.	Learn about the interoperability and compatibility factors that could affect how you upgrade to BI 12.2.1.1.	It is important to understand how two or more Oracle Fusion Middleware products of the same version or different versions work together (interoperate) in a supported Oracle Fusion Middleware configuration. You can learn more about interoperability and compatibility in <i>Understanding Interoperability and Compatibility</i> .
3	Mandatory.	Back up your existing environment.	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.

Table 3-2 (Cont.) Roadmap for Standalone Oracle HTTP Server Upgrade Process

Step No.	Importance (Mandatory, Conditional, or Optional)	Task	Description
4	Mandatory.	Complete the pre-upgrade tasks.	<p>The pre-upgrade tasks include cloning your production environment, verifying system requirements and certifications, purging unused data, and creating non-SYSDBA user.</p> <p>For a complete list of pre-upgrade tasks, see Preparing to Upgrade Oracle HTTP Server.</p>
5	Mandatory.	Install the Standalone Oracle HTTP Server.	<p>Run the installation program to install the software. Select the installation type Standalone Oracle HTTP Server (managed independently of WebLogic server). This transfers the software to your system and creates a new Oracle home directory.</p> <p>See Installing the Standalone Oracle HTTP Server.</p>
6	Mandatory.	Upgrade the 11g domain configurations with the Upgrade Assistant.	<p>After the installation, you need to use the Upgrade Assistant to upgrade the Oracle HTTP Server and system component infrastructure.</p> <p>See Upgrading the 11g Domain using Upgrade Assistant.</p>
7	Mandatory.	Complete the post-upgrade tasks and verify the upgrade.	<p>Your Oracle HTTP Server should continue to function as expected. If you have post-upgrade issues, you will need to troubleshoot the installation and retry the upgrade.</p> <p>See <i>Troubleshooting Oracle HTTP Server</i> in <i>Administering Oracle HTTP Server</i>.</p>

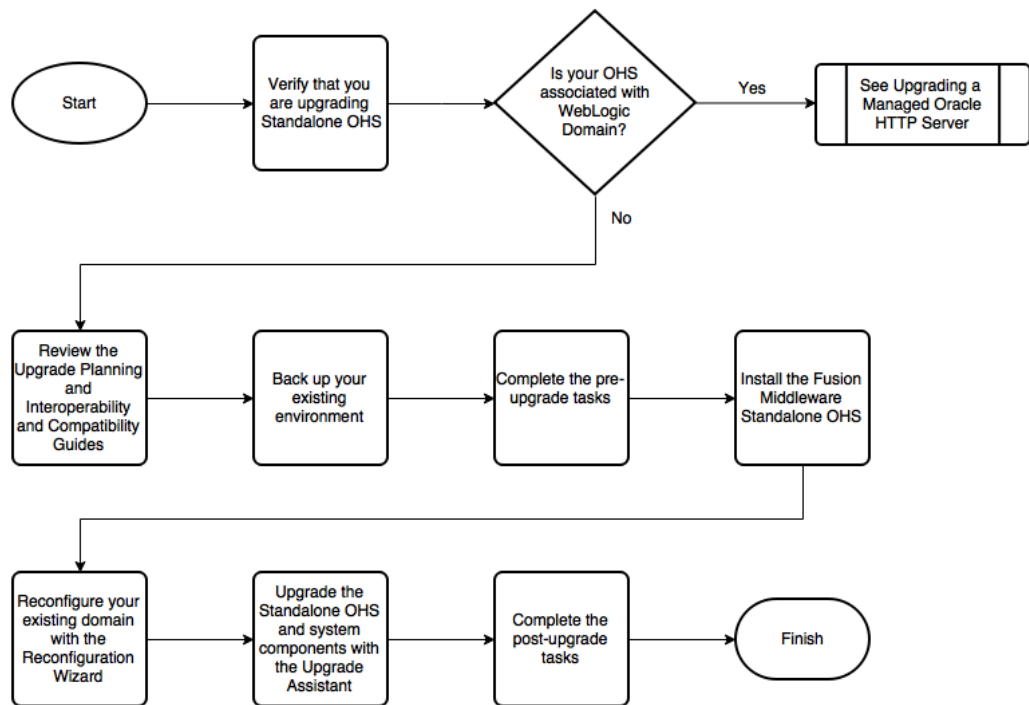
3.2.3 Flow Chart of the Standalone Oracle HTTP Server Upgrade Process from a Previous 12c Release

Figure 3-3 shows the high-level procedures associated with a standalone Oracle HTTP Server upgrade when the starting point is another 12c release. The tools used for each step are also listed.

Detailed instructions for upgrading your standalone Oracle HTTP Server are described in [Upgrading a Standalone Oracle HTTP Server from a Previous 12c Release](#).

Note: Before starting the upgrade, you must shutdown the 12c instance.

Figure 3-3 Standalone Oracle HTTP Server Upgrade from a Previous 12c release



3.2.4 Roadmap for Upgrading a Standalone Oracle HTTP Server from a Previous 12c Release

Table 3-3 describes the tasks that must be completed to upgrade a standalone Oracle HTTP Server from a previous release of 12c to the latest version.

Table 3-3 Roadmap for Standalone Oracle HTTP Server Upgrade Process

Table 3-3 (Cont.) Roadmap for Standalone Oracle HTTP Server Upgrade Process

Step No.	Importance (Mandatory, Conditional, or Optional)	Task	Description
1	Recommended.	Verify that you are upgrading a standalone Oracle HTTP Server.	To determine which Oracle HTTP Server you have in your existing environment, see Determining whether Oracle HTTP Server is Standalone or Managed (Collocated) .
2	Optional.	Learn about the interoperability and compatibility factors that could affect how you upgrade to BI 12.2.1.1.	It is important to understand how two or more Oracle Fusion Middleware products of the same version or different versions work together (interoperate) in a supported Oracle Fusion Middleware configuration. You can learn more about interoperability and compatibility in <i>Understanding Interoperability and Compatibility</i> .
3	Mandatory.	Back up your existing environment.	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.
4	Mandatory.	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 Preparing to Upgrade Oracle HTTP Server .

Table 3-3 (Cont.) Roadmap for Standalone Oracle HTTP Server Upgrade Process

Step No.	Importance (Mandatory, Conditional, or Optional)	Task	Description
5	Mandatory.	Install the Standalone Oracle HTTP Server.	Run the installation program to install the software. Select the installation type Standalone Oracle HTTP Server (managed independently of WebLogic server) . This transfers the software to your system and creates a new Oracle home directory. See Installing the Standalone Oracle HTTP Server .
6	Mandatory.	Reconfigure your existing domain.	Run the Reconfiguration Wizard from 12c Oracle HTTP Server installation to reconfigure the existing domain. See Reconfiguring the Existing Domain with the Reconfiguration Wizard .
7	Mandatory.	Upgrade the domain configurations with the Upgrade Assistant.	After the installation, you need to use the Upgrade Assistant to upgrade the Oracle HTTP Server and system component infrastructure. See Upgrading the 12c Domain using Upgrade Assistant .
8	Mandatory.	Complete the post-upgrade tasks and verify the upgrade.	Your Oracle HTTP Server should continue to function as expected. If you have post-upgrade issues, you need to troubleshoot the installation and retry the upgrade. Troubleshooting Oracle HTTP Server in <i>Administering Oracle HTTP Server</i> .

Upgrading a Standalone Oracle HTTP Server from 11g to 12c

This chapter describes the procedures for upgrading an 11g standalone Oracle HTTP Server to 12c; one that is not managed by, or registered to, an existing Oracle WebLogic Server (WLS) domain.

This chapter contains the following sections:

[Upgrading a Standalone Oracle HTTP Server](#)

[Important Pre-Upgrade Considerations](#)

Before you begin the upgrade, it is important to make sure that your existing setup is not impacted during or after the upgrade. If you are using Oracle Web Cache or WebGate, or if you have Application-specific artifacts in your 11g domain, review the topics under this section carefully to prevent impact to your existing setup.

[Upgrading an 11g Standalone Oracle HTTP Server](#)

Upgrading a standalone Oracle HTTP Server from 11g to 12c is an out-of-place upgrade. You must install the Oracle HTTP Server 11g software in a new Oracle home and then use the Upgrade Assistant to upgrade the 11g domain.

4.1 Important Pre-Upgrade Considerations

Before you begin the upgrade, it is important to make sure that your existing setup is not impacted during or after the upgrade. If you are using Oracle Web Cache or WebGate, or if you have Application-specific artifacts in your 11g domain, review the topics under this section carefully to prevent impact to your existing setup.

[Oracle Web Cache 11g Users](#)

Oracle Web Cache is a secure reverse proxy cache and a compression engine deployed between a browser and the HTTP server or a browser and the Content Management server to improve the performance of the websites by caching frequently accessed content. Oracle has released the last version of Web Cache in 11g. Web Cache is not available in 12c.

[WebGate 11g Users](#)

A WebGate is a web-server plug-in for Oracle Access Manager (OAM) that intercepts HTTP requests and forwards them to the Access Server for authentication and authorization. WebGate is included as part of Oracle HTTP Server 12c installation and is upgraded as part of the Oracle HTTP Server upgrade process through Upgrade Assistant.

[Updating the WebGate log file-name](#)

In Oracle Fusion Middleware 12c Release 12.2.1, WebGate log file is renamed from oblog.log to webgate.log. After upgrading to the latest

release of Oracle HTTP Server, you must manually update the `oblog_config_wg.xml` file and replace every occurrence of `'oblog.log'` with `'webgate.log'`.

Application Artifacts from 11g

Application artifacts include all of your web resources: JSP files, images, stylesheets, Javascript, static HTML pages in addition to your Java classes and source files and web application configuration files. The integrated development environment (IDE) uses all of these resources, and refers to them as web application artifacts.

Migrating 11g Application Artifacts

4.1.1 Oracle Web Cache 11g Users

Oracle Web Cache is a secure reverse proxy cache and a compression engine deployed between a browser and the HTTP server or a browser and the Content Management server to improve the performance of the websites by caching frequently accessed content. Oracle has released the last version of Web Cache in 11g. Web Cache is not available in 12c.

Consider the following limitations if you are using Web Cache in your 11g environment:

- Web Cache is not available in Fusion Middleware 12c. Correspondingly, there is no upgrade for Web Cache.
- Web Cache 11g front-ending a 12c Oracle HTTP Server is not a certified combination.
- If you are using both Web Cache and Oracle HTTP Server in your 11g setup, you can only upgrade the Oracle HTTP Server to 12c. In that case, you need to disable the 11g Web Cache and change the configuration settings to divert the traffic to Oracle HTTP Server directly.
- If you are using both Web Cache and Oracle HTTP Server in your 11g setup, and you have registered them to a Oracle WebLogic Server (WLS) domain for administering them through the Enterprise Manager Fusion Middleware Control, then you must also upgrade the WLS domain to 12c. In that case, the associated Oracle HTTP Server is also upgraded to 12c. However, the Web Cache is removed from the 12c Fusion Middleware Control.

4.1.2 WebGate 11g Users

A WebGate is a web-server plug-in for Oracle Access Manager (OAM) that intercepts HTTP requests and forwards them to the Access Server for authentication and authorization. WebGate is included as part of Oracle HTTP Server 12c installation and is upgraded as part of the Oracle HTTP Server upgrade process through Upgrade Assistant.

Note: In Oracle Fusion Middleware 12c Release 12.2.1.1, WebGate log file is renamed from `oblog.log` to `webgate.log`. After upgrading to the latest release of Oracle HTTP Server, you must manually update the `oblog_config_wg.xml` file and replace every `'oblog.log'` occurrence with `'webgate.log'`. For a procedure to do this task, see [Updating the WebGate log file-name](#).

4.1.3 Updating the WebGate log file-name

In Oracle Fusion Middleware 12c Release 12.2.1, WebGate log file is renamed from `oblog.log` to `webgate.log`. After upgrading to the latest release of Oracle HTTP Server, you must manually update the `oblog_config_wg.xml` file and replace every occurrence of `'oblog.log'` with `'webgate.log'`.

To update the WebGate log file-name:

1. Change directory to the following path:

```
DOMAIN_HOME/config/fmwconfig/components/OHS/ohs_instance/  
webgate/config
```

2. Open the `oblog_config_wg.xml` file in edit mode.
3. Replace every occurrence of `'oblog.log'` with `'webgate.log'`.

4.1.4 Application Artifacts from 11g

Application artifacts include all of your web resources: JSP files, images, stylesheets, Javascript, static HTML pages in addition to your Java classes and source files and web application configuration files. The integrated development environment (IDE) uses all of these resources, and refers to them as web application artifacts.

If you have 11g application artifacts that you want to continue using in 12c, carefully review the following:

- As part of upgrading Oracle HTTP Server from an 11g Oracle instance to a 12c domain, the Oracle HTTP Server configuration directory layout is being migrated from an Oracle instance to a standalone domain.
- Oracle HTTP Server 11g configuration files that reside in the component configuration directory of the Oracle instance are migrated automatically.
- Application artifacts that reside within the Oracle instance, including any combination of static files (such as HTML or images, CGI or FastCGI scripts or applications, or third-party modules), must be migrated manually after the upgrade to 12c.

For more information, see [Migrating 11g Application Artifacts](#).

4.1.5 Migrating 11g Application Artifacts

You will have to manually migrate any 11g application artifacts that reside within the Oracle instance, including any combination of static files such as HTML or images, CGI or FastCGI scripts or applications, or third-party modules. Application artifacts that were referred to by the 11g configuration, but were stored in directories outside of the Oracle instance, will continue to be referenced by the migrated configuration used by Oracle HTTP Server 12c.

For example, if a third-party plug-in module was installed into the Oracle home with Oracle HTTP Server 11g, and the configuration referenced it by the Oracle home location using the configuration in the example below, the plug-in module must be installed manually into the Oracle home with 12c Oracle HTTP Server or the upgraded configuration must be modified to reference it elsewhere.

```
LoadModule example_module "${ORACLE_HOME}/ohs/modules/mod_example.so"
```

Similarly, if static files were copied into the `/htdocs` directory within the 11g component configuration directory, then they too must be manually copied into the 12c instance configuration directory or the upgraded configuration must be modified to reference it elsewhere. Other types of application artifacts must be manually migrated in a similar manner.

4.2 Upgrading an 11g Standalone Oracle HTTP Server

Upgrading a standalone Oracle HTTP Server from 11g to 12c is an out-of-place upgrade. You must install the Oracle HTTP Server 11g software in a new Oracle home and then use the Upgrade Assistant to upgrade the 11g domain.

Perform the following tasks to complete the standalone Oracle HTTP Server upgrade:

[Installing the Standalone Oracle HTTP Server](#)

[Upgrading the 11g Domain using Upgrade Assistant](#)

[Verifying the Upgrade](#)

4.2.1 Installing the Standalone Oracle HTTP Server

To install the standalone Oracle HTTP Server:

1. Sign in to the target system where you want to install the 12.2.1.1 product distribution.
2. Download the Oracle HTTP Server distribution (`fmw_12.2.1.1.0_ohs_linux64.bin` for UNIX and `setup_fmw_12.2.1.1.0_ohs_win64.exe` for Windows) from [Oracle Technology Network](#) or [Oracle Software Delivery Cloud](#) on your target system.
3. Change to the directory where you downloaded the Oracle HTTP Server distribution.
4. Check that your machines meet the following requirements:
 - Ensure that the system, patch, kernel, and other requirements are met as specified in *Installing and Configuring Oracle HTTP Server*.
 - Because Oracle HTTP Server is installed by default on port 7777, you must ensure that port 7777 is not used by any service on the nodes. To check if this port is in use, run the following command before installing Oracle HTTP Server. You must free the port if it is in use.

```
netstat -an | grep 7777
```

5. Start the installation program by entering the following command:

On UNIX operating system:

```
./ohs_linux64.bin
```

On Windows operating system:

```
setup_ohs_win64.exe
```

6. 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.

7. On the Welcome screen, review the information to make sure that you have met all the prerequisites and click **Next**.
8. 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.
9. On the Installation Location screen, specify the location for the Oracle home directory and click **Next**.

Since you are installing the standard installation topology for Oracle HTTP Server in a standalone domain, you can specify an Oracle home directory of your choice. However, ensure that you install the software in a new Oracle home.

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

10. On the Installation Type screen, select **Standalone HTTP Server (Managed independently of WebLogic server)** and click **Next**.

For more information about the installation types, see *Introducing the Oracle HTTP Server Standard Installation Topologies in Installing and Configuring Oracle HTTP Server*.

11. 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.

12. On the Security Updates screen, indicate how you would like to receive security updates if you already have an Oracle Support account.

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.

13. 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**.

14. On the Installation Progress screen, click **Next** when the progress bar displays 100%.
15. 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.

You have installed the Oracle HTTP Server in a standalone mode.

4.2.2 Upgrading the 11g Domain using Upgrade Assistant

To upgrade the 11g domain using Upgrade Assistant:

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 Domain Directory screen, select **Create a New Domain (choose when upgrading from 11g)**.
4. On the Standalone Components screen, select **Standalone System Component Configurations**.

Select **Create a New Domain** and click **Next**.

Note:

- **Create a New Domain** option should only be used if you are upgrading from version 11g.
 - If you have already created a new 12c standalone domain for your upgraded 11g system components, you can extend the standalone domain with the standalone Oracle HTTP Server using **Update an Existing Domain** option.
 - **Update an Existing Domain** option should only be used if you are upgrading from a previous 12c release (12.1.2, 12.1.3 or 12.2.1.0) to the latest 12c release (12.2.1.1).
-
-

5. 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.

6. 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.

7. On the Instance Directories screen, select **11g Source** and provide the directory location of the 11g instance that you want to upgrade.

This instance is used as a basis for configuring the 12c Oracle HTTP Server instance. The 11g Oracle HTTP Server installation is not altered.

Click **+** to add additional 11g instance directories, if necessary. Click **Next**.

Note: You cannot use the Upgrade Assistant to upgrade Oracle 10g instances to Oracle 12c. You must first upgrade Oracle 10g instances to 11g. For more information on migrating 10g to 11g, see the 11g upgrade documentation for your components.

8. On the Node Manager screen, specify the following to create a new Node Manager to administer your standalone domain.

Table 4-1 Field Descriptions for the Node Manager Screen

Option	Description
User Name	To specify the user name used to access Node Manager.
Password	To specify the password used to access Node Manager. Re-enter the password for confirmation.
Listen Address	To enter the DNS name or IP address upon which Node Manager listens.
Listen Port	To enter the listening port number of Node Manager.

Note: The user name and password are only used to authenticate connections between Node Manager and clients. They are independent from the server Administrator credentials.

9. 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.
-
-

- 10.** 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.

- 11.** 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.

- 12.** 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.

- 13.** 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.

4.2.3 Verifying the Upgrade

You can verify that the upgrade is successful if you are able to start the Node Manager and the Standalone Oracle HTTP Server properly. If you experience post-upgrade issues, you need to troubleshoot the installation and retry the upgrade. For more information, see Troubleshooting Oracle HTTP Server in *Administrator's Guide for Oracle HTTP Server*.

To start the Node Manager and the Standalone Oracle HTTP Server, see the following topics:

Starting the Node Manager

Server instances in a WebLogic Server production environment are often distributed across multiple domains, machines, and geographic locations. Node Manager is a WebLogic Server utility that enables you to manage (start, shut down, and restart) your Oracle HTTP Server instance.

Starting the Standalone Oracle HTTP Server

You can use the `startComponent.sh | cmd` script to start the standalone Oracle HTTP Server.

4.2.3.1 Starting the Node Manager

Server instances in a WebLogic Server production environment are often distributed across multiple domains, machines, and geographic locations. Node Manager is a WebLogic Server utility that enables you to manage (start, shut down, and restart) your Oracle HTTP Server instance.

On Windows operating systems, it is recommended that you configure Node Manager to run as a startup service. This allows Node Manager to start up automatically each time the system is restarted. For more information, see *Running Node Manager as a Startup Service* in *Administering Node Manager for Oracle WebLogic Server*.

Note: On UNIX platform, do not run Node Manager as the root user.

1. Change directory to the following:

On UNIX operating system:

```
$DOMAIN_HOME/bin
```

On Windows operating system:

```
%DOMAIN_HOME%\bin
```

2. On the UNIX operating system, start the Node Manager by running the following command:

```
nohup ./startNodeManager.sh > nm.out&
```

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

On the Windows operating system, start the Node Manager by running the following command:

```
startNodeManager.cmd
```

You can also use a shortcut on the **Start** menu to start the Node Manager. Click **Tools** and then click **Node Manager**.

4.2.3.2 Starting the Standalone Oracle HTTP Server

You can use the `startComponent.sh | cmd` script to start the standalone Oracle HTTP Server.

To start the standalone Oracle HTTP Server:

1. Change directory to the following

On UNIX operating system:

```
$DOMAIN_HOME/bin
```

On Windows operating system:

```
%DOMAIN_HOME%\bin
```

2. On UNIX operating system, enter the following command:

```
./startComponent.sh ohs_name
```

On 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*.

Upgrading a Standalone Oracle HTTP Server from a Previous 12c Release

This chapter describes the process of upgrading a Standalone Oracle HTTP Server to the latest 12c from a previous Standalone Oracle HTTP Server 12c release.

Note:

The information in this chapter assumes that you have read and performed any required pre-upgrade tasks in *Planning an Upgrade of Oracle Fusion Middleware*.

This chapter contains the following sections:

[Upgrading a Standalone Oracle HTTP Server](#)

[Installing the Standalone Oracle HTTP Server](#)

[Reconfiguring the Existing 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 12c Domain using Upgrade Assistant](#)

[Verifying the Upgrade](#)

5.1 Installing the Standalone Oracle HTTP Server

To install the standalone Oracle HTTP Server:

1. Sign in to the target system where you want to install the 12.2.1.1 product distribution.
2. Download the Oracle HTTP Server distribution (fmw_12.2.1.1.0_ohs_linux64.bin for UNIX and setup_fmw_12.2.1.1.0_ohs_win64.exe for Windows) from [Oracle Technology Network](#) or [Oracle Software Delivery Cloud](#) on your target system.
3. Change to the directory where you downloaded the Oracle HTTP Server distribution.
4. Check that your machines meet the following requirements:
 - Ensure that the system, patch, kernel, and other requirements are met as specified in *Installing and Configuring Oracle HTTP Server*.

- Because Oracle HTTP Server is installed by default on port 7777, you must ensure that port 7777 is not used by any service on the nodes. To check if this port is in use, run the following command before installing Oracle HTTP Server. You must free the port if it is in use.

```
netstat -an | grep 7777
```

5. Start the installation program by entering the following command:

On UNIX operating system:

```
./ohs_linux64.bin
```

On Windows operating system:

```
setup_ohs_win64.exe
```

6. 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.

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

8. 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.

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

Since you are installing the standard installation topology for Oracle HTTP Server in a standalone domain, you can specify an Oracle home directory of your choice. However, ensure that you install the software in a new Oracle home.

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

10. On the Installation Type screen, select **Standalone HTTP Server (Managed independently of WebLogic server)** and click **Next**.

For more information about the installation types, see *Introducing the Oracle HTTP Server Standard Installation Topologies* in *Installing and Configuring Oracle HTTP Server*.

11. 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.

12. On the Security Updates screen, indicate how you would like to receive security updates if you already have an Oracle Support account.

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.

13. 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**.

14. On the Installation Progress screen, click **Next** when the progress bar displays 100%.

15. 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.

You have installed the Oracle HTTP Server in a standalone mode.

5.2 Reconfiguring the Existing 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.

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 5-1 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> .

Table 5-1 (Cont.) Field Descriptions for Node Manager Screen

Option	Description
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.
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.

5.3 Upgrading the 12c Domain using Upgrade Assistant

To upgrade the 12c domain using Upgrade Assistant:

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 Standalone Components screen, select **Standalone System Component Configurations**.

Select **Update an Existing Domain** and enter the location of the existing 12c domain in the **Domain Directory** field. You can also click **Browse** to select the existing domain directory using the navigation tree. Click **Next**.

Note:

- **Create a New Domain** option should only be used if you are upgrading from version 11g.
 - If you have already created a new 12c standalone domain for your upgraded 11g system components, you can extend the standalone domain with the standalone Oracle HTTP Server using **Update an Existing Domain** option.
 - **Update an Existing Domain** option should only be used if you are upgrading from a previous 12c release (12.1.2, 12.1.3 or 12.2.1.0) to the latest 12c release (12.2.1.1).
-

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 Instance Directories screen, select **12c Source** and click **Next**.

You do not need to enter the Oracle instance directories when upgrading to the latest 12c release from a previous 12c release.

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.

5.4 Verifying the Upgrade

You can verify that the upgrade is successful if you are able to start the Node Manager and the Standalone Oracle HTTP Server properly. If you experience post-upgrade issues, you need to troubleshoot the installation and retry the upgrade. For more information, see Troubleshooting Oracle HTTP Server in *Administrator's Guide for Oracle HTTP Server*.

To start the Node Manager and the Standalone Oracle HTTP Server, see the following topics:

[Starting the Node Manager](#)

Server instances in a WebLogic Server production environment are often distributed across multiple domains, machines, and geographic

locations. Node Manager is a WebLogic Server utility that enables you to manage (start, shut down, and restart) your Oracle HTTP Server instance.

Starting the Standalone Oracle HTTP Server

You can use the `startComponent.sh | cmd` script to start the standalone Oracle HTTP Server.

5.4.1 Starting the Node Manager

Server instances in a WebLogic Server production environment are often distributed across multiple domains, machines, and geographic locations. Node Manager is a WebLogic Server utility that enables you to manage (start, shut down, and restart) your Oracle HTTP Server instance.

On Windows operating systems, it is recommended that you configure Node Manager to run as a startup service. This allows Node Manager to start up automatically each time the system is restarted. For more information, see *Running Node Manager as a Startup Service* in *Administering Node Manager for Oracle WebLogic Server*.

Note: On UNIX platform, do not run Node Manager as the root user.

1. Change directory to the following:

On UNIX operating system:

```
$DOMAIN_HOME/bin
```

On Windows operating system:

```
%DOMAIN_HOME%\bin
```

2. On the UNIX operating system, start the Node Manager by running the following command:

```
nohup ./startNodeManager.sh > nm.out&
```

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

On the Windows operating system, start the Node Manager by running the following command:

```
startNodeManager.cmd
```

You can also use a shortcut on the **Start** menu to start the Node Manager. Click **Tools** and then click **Node Manager**.

5.4.2 Starting the Standalone Oracle HTTP Server

You can use the `startComponent.sh | cmd` script to start the standalone Oracle HTTP Server.

To start the standalone Oracle HTTP Server:

1. Change directory to the following

On UNIX operating system:

`$DOMAIN_HOME/bin`

On Windows operating system:

`%DOMAIN_HOME%\bin`

2. On UNIX operating system, enter the following command:

`./startComponent.sh ohs_name`

On 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*.

Part II

Upgrading a Managed Oracle HTTP Server

Part II contains the following chapters:

[Introduction to the Managed Oracle HTTP Server Upgrade](#)

This chapter provides a general introduction to the Oracle Fusion Middleware Managed Oracle HTTP Server and an overview of the steps you must perform to upgrade from an Oracle Fusion Middleware 11g or a previous 12c Release (12.1.2 and above) of Oracle HTTP Server to this Oracle HTTP Server 12c Release (12.2.1.1).

[Upgrading a Managed Oracle HTTP Server from 11g to 12c](#)

This chapter describes the procedure for upgrading a managed Oracle HTTP Server from 11g to 12c. The valid starting points for this upgrade are 11g releases 11.1.1.7 and above.

[Upgrading a Managed Oracle HTTP Server from a Previous 12c Release](#)

This chapter describes the procedure for upgrading a managed Oracle HTTP Server from to a latest 12c release from a previous 12c release. The valid starting point for this upgrade are 12c releases 12.1.2 and above.

Introduction to the Managed Oracle HTTP Server Upgrade

This chapter provides a general introduction to the Oracle Fusion Middleware Managed Oracle HTTP Server and an overview of the steps you must perform to upgrade from an Oracle Fusion Middleware 11g or a previous 12c Release (12.1.2 and above) of Oracle HTTP Server to this Oracle HTTP Server 12c Release (12.2.1.1).

This chapter includes the following sections:

[Upgrading a Managed Oracle HTTP Server](#)

[Flowchart of Managed Oracle HTTP Server Upgrade Process](#)

This topic helps you to understand the process flow of the Oracle HTTP Server upgrade from a 11g or a previous 12c release using a flowchart.

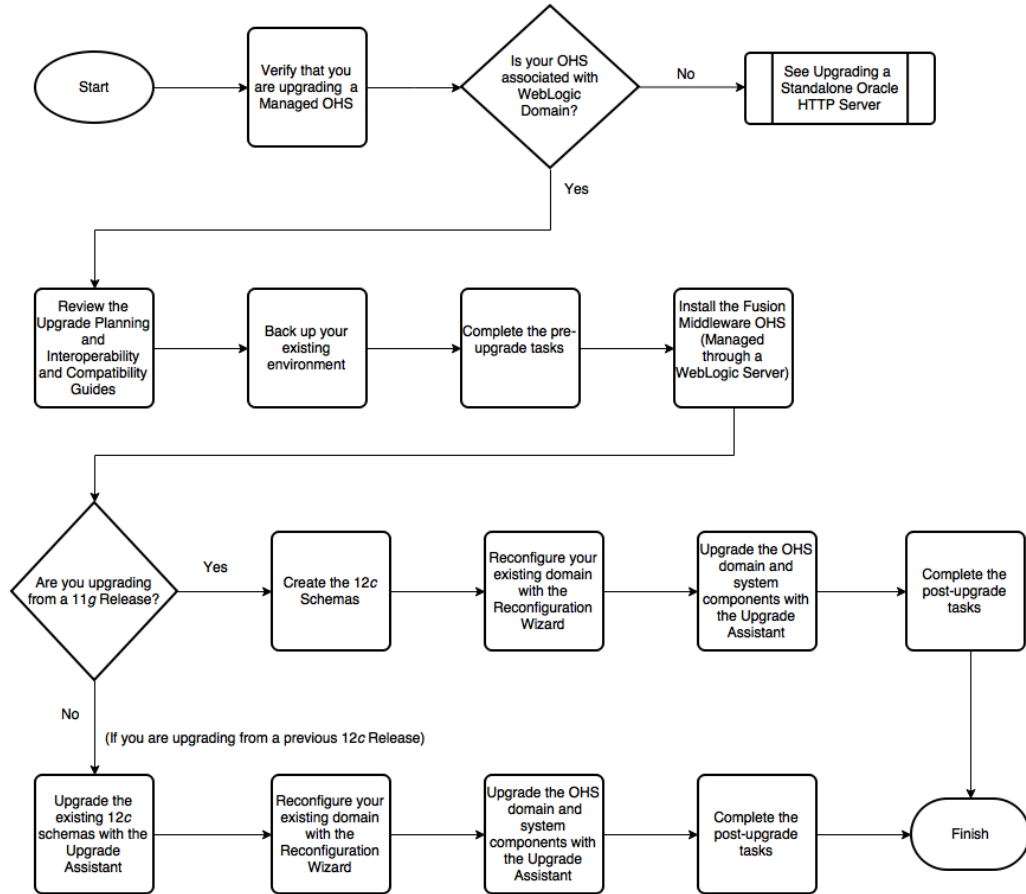
[Roadmap for Upgrading a Managed Oracle HTTP Server](#)

6.1 Flowchart of Managed Oracle HTTP Server Upgrade Process

This topic helps you to understand the process flow of the Oracle HTTP Server upgrade from a 11g or a previous 12c release using a flowchart.

[Figure 6-1](#) shows the process flow to upgrade a Managed Oracle HTTP Server to the latest 12c release.

Figure 6-1 Flowchart for Managed Oracle HTTP Server Upgrade Process



6.2 Roadmap for Upgrading a Managed Oracle HTTP Server

Table 6-1 describes the tasks that must be completed to upgrade a Managed Oracle HTTP Server from a 11g release or a previous 12c release to the latest version (12.2.1.1).

Table 6-1 Roadmap for Managed Oracle HTTP Server Upgrade Process

Step No.	Importance (Mandatory, Conditional, or Optional)	Task	Description
1	Recommended.	Verify that you are upgrading a Managed Oracle HTTP Server.	To determine which Oracle HTTP Server you have in your existing environment, see Determining whether Oracle HTTP Server is Standalone or Managed (Collocated) .

Table 6-1 (Cont.) Roadmap for Managed Oracle HTTP Server Upgrade Process

Step No.	Importance (Mandatory, Conditional, or Optional)	Task	Description
2	Optional.	Learn about the interoperability and compatibility factors that could affect how you upgrade to BI 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>You can learn more about interoperability and compatibility in <i>Understanding Interoperability and Compatibility</i>.</p>
3	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>
4	Mandatory.	Complete the pre-upgrade tasks.	<p>The pre-upgrade tasks include cloning your production environment, verifying system requirements and certifications, purging unused data, and creating non-SYSDBA user.</p> <p>For a complete list of pre-upgrade tasks, see Preparing to Upgrade Oracle HTTP Server.</p>

Table 6-1 (Cont.) Roadmap for Managed Oracle HTTP Server Upgrade Process

Step No.	Importance (Mandatory, Conditional, or Optional)	Task	Description
5	Mandatory.	Install the Managed Oracle HTTP Server.	Run the installation program to install the software. Select the installation type Managed Oracle HTTP Server (managed independently of WebLogic server) . This transfers the software to your system and creates a new Oracle home directory. See Installing the Managed 12c (12.2.1.1.0) Oracle HTTP Server .
6	Conditional: If you are upgrading from a 11g release.	Create the required 12c schemas.	See Creating the Required Schemas before Upgrade .
7	Conditional: If you are upgrading from a previous 12c release.	Upgrade the existing schemas with the Upgrade Assistant.	See Upgrading the Existing 12c Schema with the Upgrade Assistant .
8	Mandatory.	Reconfigure your existing domain.	Run the Reconfiguration Wizard from 12c Oracle HTTP Server installation to reconfigure the existing domain. See Reconfiguring the Existing Domain with the Reconfiguration Wizard .
9	Mandatory.	Upgrade the domain configurations with the Upgrade Assistant.	After the installation, you need to use the Upgrade Assistant to upgrade the Oracle HTTP Server and system component infrastructure. See Upgrading the Component Configuration with the Upgrade Assistant .
10	Mandatory.	Complete the post-upgrade tasks and verify the upgrade.	For 11g to 12c post-upgrade tasks, see Post-Upgrade Tasks . For previous 12c post-upgrade tasks, see Post-Upgrade Tasks .

Upgrading a Managed Oracle HTTP Server from 11g to 12c

This chapter describes the procedure for upgrading a managed Oracle HTTP Server from 11g to 12c. The valid starting points for this upgrade are 11g releases 11.1.1.7 and above.

Upgrading a Managed Oracle HTTP Server

Installing the Managed 12c (12.2.1.1.0) Oracle HTTP Server

Before you upgrade your existing Managed Oracle HTTP Server 12c (12.1.2 or 12.1.3 or 12.2.1) components, you must first install the Oracle HTTP Server Release 12.2.1.1.0.

Creating the Required Schemas before Upgrade

If you are upgrading from Oracle HTTP Server 11g, you must create the required 12c schemas before you begin the upgrade. The schemas required for Oracle HTTP Server are: Service Table (STB) and Oracle Platform Security Services (OPSS).

Reconfiguring the 11g Domain with the Reconfiguration Wizard

You need to upgrade the 11g Oracle HTTP Server domain using the 12c Reconfiguration Wizard.

Upgrading the Component Configuration with the Upgrade Assistant

If you are running the Upgrade Assistant from an Oracle home that contains managed domain components, then the **WebLogic Component Configuration** upgrade option is available.

Post-Upgrade Tasks

The post-upgrade tasks include starting the Node Manager, starting the Administration Server, starting the Oracle HTTP Server, and verifying whether the upgrade from 11g to 12c is successful.

7.1 Installing the Managed 12c (12.2.1.1.0) Oracle HTTP Server

Before you upgrade your existing Managed Oracle HTTP Server 12c (12.1.2 or 12.1.3 or 12.2.1) components, you must first install the Oracle HTTP Server Release 12.2.1.1.0.

To install the Oracle HTTP Server:

1. Log in to the host where you want to install the Oracle HTTP Server.
2. Download the Oracle HTTP Server product distribution (fmw_12.2.1.1.0_ohs_linux64.bin | .exe) from [Oracle Technology Network](#) or [Oracle Software Delivery Cloud](#) on your host system.
3. Check that your machines meet the following requirements:

- Ensure that the system, patch, kernel, and other requirements are met as specified in *Installing and Configuring Oracle HTTP Server*.
- Because Oracle HTTP Server is installed by default on port 7777, you must ensure that port 7777 is not used by any service on the nodes. To check if this port is in use, run the following command before installing Oracle HTTP Server. You must free the port if it is in use.

```
netstat -an | grep 7777
```

4. On UNIX platforms, if the `/etc/oraInst.loc` file exists, check that its contents are correct. Specifically, check that the inventory directory is correct and that you have write permissions for that directory.

If the `/etc/oraInst.loc` file does not exist, you can skip this step.

5. The Oracle HTTP Server distribution is a `.bin` file. Run the installer by entering the following command:

```
./distribution_filename.bin
```

6. On the Installation Inventory Setup screen, specify the location where you want to create your central inventory in the **Inventory Directory** field.

Make sure that the Operating System Group name that you select on this screen has write permissions to the central inventory location. Click **OK**.

Note: This screen does not appear on Windows operating systems. For more information about the central inventory, see *Understanding the Oracle Central Inventory in Installing Software with the Oracle Universal Installer*.

7. On the Welcome screen, review the information and click **Next**.
8. On the Auto Updates screen, select one of the following 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.
9. On the Installation Location screen, specify the location for the Oracle home directory and click **Next**.

Since you are installing the standard installation topology for a collocated Oracle HTTP Server in a WebLogic Server domain, enter the path to an existing Oracle Fusion Middleware Infrastructure Oracle home.

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

10. On the Installation Type screen, select **Collocated HTTP Server (Managed through WebLogic Server)** and click **Next**.

For more information about the installation types, see Introducing the Oracle HTTP Server Standard Installation Topologies in *Installing and Configuring Oracle HTTP Server*.

11. On the Prerequisite Check screen, verify that your system meets the minimum necessary requirements and click **Next**. If you see a warning or error message, see Oracle Fusion Middleware System Requirements and Specifications.

For more information about other options on this screen, click **Help**.

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%.

If the installation is successful, the Installation Complete screen displays the message Oracle HTTP Server (OHS) installation completed successfully. Click **Finish** to dismiss the installer.

You have installed the Oracle HTTP Server in a collocated mode.

7.2 Creating the Required Schemas before Upgrade

If you are upgrading from Oracle HTTP Server 11g, you must create the required 12c schemas before you begin the upgrade. The schemas required for Oracle HTTP Server are: Service Table (STB) and Oracle Platform Security Services (OPSS).

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*.

To create the 12c schema:

1. Change directory to the following:

```
12c_Oracle_Home/oracle_common/bin/
```

2. Run the RCU by entering the following command:

```
./rcu
```

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

Table 7-1 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.exempldomain.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.exempldomain.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 7-2 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 7-3 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 7-3 (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 7-4 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.

The custom prefix is used to logically group these schemas together for use in this domain.

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

You must remember the prefix and schema names for the components you are installing; you need this information while configuring your product installation. Oracle recommends that you note these values.

- Verify that the prerequisites checking is successful. Click **OK** to go to the next page.
- 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.

10. 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.

11. 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](#)

7.3 Reconfiguring the 11g Domain with the Reconfiguration Wizard

You need to upgrade the 11g Oracle HTTP Server domain using the 12c Reconfiguration Wizard.

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 7-5 Field Descriptions for Node Manager Screen

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

Table 7-5 (Cont.) Field Descriptions for Node Manager Screen

Option	Description
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 7-5 (Cont.) Field Descriptions for Node Manager Screen

Option	Description
Apply Oracle Recommended Defaults	<p>If you selected the Migrate Existing Configuration option, select this check box if you want to use Oracle-recommended defaults in the 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.

7.4 Upgrading the Component Configuration with the Upgrade Assistant

If you are running the Upgrade Assistant from an Oracle home that contains managed domain components, then the **WebLogic Component Configuration** upgrade option is available.

To upgrade the component configurations for Oracle HTTP Server:

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 WebLogic Components screen, select the **WebLogic Component Configurations** option to upgrade component configurations for a managed WebLogic Server domain. Enter the connection details required to connect to the WebLogic Administration Server that is managing the domain and 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. 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.
-
-

7. 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.

8. 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.

9. 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.

10. 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.

7.5 Post-Upgrade Tasks

The post-upgrade tasks include starting the Node Manager, starting the Administration Server, starting the Oracle HTTP Server, and verifying whether the upgrade from 11g to 12c is successful.

Starting the Node Manager

Server instances in a WebLogic Server production environment are often distributed across multiple domains, machines, and geographic locations. Node Manager is a WebLogic Server utility that enables you to start, shut down, and restart Administration Server and Managed Server instances from a remote location.

Starting the Administration Server

One instance of WebLogic Server in each domain acts as an 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 Oracle HTTP Server

You can use Enterprise Manager Fusion Middleware Control or the `opmnctl` command to start, stop, and restart Oracle HTTP Server.

Verifying that Oracle HTTP Server Installation is Successful

If you can log in to the Administration Console and the Enterprise Manager Fusion Middleware Control, you have installed Oracle HTTP Server successfully.

7.5.1 Starting the Node Manager

Server instances in a WebLogic Server production environment are often distributed across multiple domains, machines, and geographic locations. Node Manager is a WebLogic Server utility that enables you to start, shut down, and restart Administration Server and Managed Server instances from a remote location.

On Windows operating systems, it is recommended that you configure Node Manager to run as a startup service. This allows Node Manager to start up automatically each time the system is restarted. For more information, see *Running Node Manager as a Startup Service* in *Administering Node Manager for Oracle WebLogic Server*.

Note: On UNIX platform, do not run Node Manager as the root user.

To start the Node Manager:

1. Change directory to the following:

On UNIX operating system:

```
$ORACLE_HOME/wlserver/server/bin/
```

On Windows operating system:

```
%ORACLE_HOME%\wlserver\server\bin\
```

2. On the UNIX operating system, start the Node Manager by running the following command:

```
nohup ./startNodeManager.sh > nm.out&
```

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

On the Windows operating system, start the Node Manager by running the following command:

```
startNodeManager.cmd
```

You can also use a shortcut on the **Start** menu to start the Node Manager. Click **Tools** and then click **Node Manager**.

7.5.2 Starting the Administration Server

One instance of WebLogic Server in each domain acts as an 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.

In a development environment, it is usually sufficient to start an Administration Server and deploy your applications directly onto the Administration Server. However, in a production environment, it is recommended that you create Managed Servers to run your applications.

To start an Administration Server:

1. Change directory to the following:

```
$DOMAIN_HOME/DOMAIN_NAME
```

Where, *DOMAIN_NAME* is the name of the directory in which you located the domain.

2. On the Windows operating system, start the Administration Server by running the following command:

```
startWebLogic.cmd
```

On the Windows operating system, the Configuration Wizard creates a shortcut on the Start menu to start the Administration Server that you created.

On the UNIX operating system, start the Administration Server by running the following command:

```
./startWebLogic.sh
```

Note: If the server prompts you to enter a username and password, enter the name of a WebLogic Server user who has permission to start servers.

7.5.3 Starting the Oracle HTTP Server

You can use Enterprise Manager Fusion Middleware Control or the `opmnctl` command to start, stop, and restart Oracle HTTP Server.

Starting the Oracle HTTP Server using the Enterprise Manager Fusion Middleware Control

To start an Oracle HTTP Server using the Enterprise Manager Fusion Middleware Control:

1. Go to the Oracle HTTP Server home page.
 2. From the Oracle HTTP Server menu:
 - a. Select **Control**.
 - b. Select **Start Up** from the **Control** menu.
- Or
3. From the Target Navigation tree:
 - a. Right-click the Oracle HTTP Server instance you want to start.
 - b. Select **Control**.
 - c. Select **Start Up** from the **Control** menu.

Starting the Oracle HTTP Server using `opmnctl`

To start Oracle HTTP Server components in an Oracle instance using `opmnctl`:

1. Change directory to the following:

```
ORACLE_INSTANCE/bin
```
2. Run the following command to start all Oracle HTTP Server components in an Oracle instance

```
opmnctl startproc process-type=OHS
```

Determining the Status of Oracle HTTP Server Components using `opmnctl`

To determine the status of the Oracle HTTP Server components:

1. Change directory to the following:

```
ORACLE_INSTANCE/bin
```
2. Enter the following command:

```
opmnctl status
```

Following is a sample output:

Processes in Instance: instance1

ias-component	process-type	pid	status
webcache1	WebCache-admin	19556	Alive
webcache1	WebCache	19555	Alive
ohs1	OHS	7249	Alive

7.5.4 Verifying that Oracle HTTP Server Installation is Successful

If you can log in to the Administration Console and the Enterprise Manager Fusion Middleware Control, you have installed Oracle HTTP Server successfully.

To verify whether the software patch is installed successfully:

1. Log in to the following URL:

<http://MachineName.us.oracle.com:7777>

2. To access Enterprise Manager Fusion Middleware Control:

<http://MachineName.us.oracle.com:7001/em>

Upgrading a Managed Oracle HTTP Server from a Previous 12c Release

This chapter describes the procedure for upgrading a managed Oracle HTTP Server from to a latest 12c release from a previous 12c release. The valid starting point for this upgrade are 12c releases 12.1.2 and above.

This procedure assumes that you have the 12.1.2 Fusion Middleware Infrastructure (JRF plus the WebLogic Server) installed and configured on your system.

Before installing the 12c software, shut down all the 12.1.2 server instances except the RDBMS.

Upgrading a Managed Oracle HTTP Server

Installing the Managed 12c (12.2.1.1.0) Oracle HTTP Server

Before you upgrade your existing Managed Oracle HTTP Server 12c (12.1.2 or 12.1.3 or 12.2.1) components, you must first install the Oracle HTTP Server Release 12.2.1.1.0.

Upgrading the Existing 12c Schema with the Upgrade Assistant

You need to upgrade the existing 12c OPSS (Oracle Platform Security Services) and IAU (Audit Services) schema using the 12c Upgrade Assistant.

Reconfiguring the Existing 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 Component Configuration with the Upgrade Assistant

If you are running the Upgrade Assistant from an Oracle home that contains managed domain components, then the **WebLogic Component Configuration** upgrade option is available.

Post-Upgrade Tasks

The post-upgrade tasks include starting the Node Manager, starting the Administration Server, starting the Oracle HTTP Server, and verifying whether the upgrade from previous 12c release to this release is successful.

8.1 Installing the Managed 12c (12.2.1.1.0) Oracle HTTP Server

Before you upgrade your existing Managed Oracle HTTP Server 12c (12.1.2 or 12.1.3 or 12.2.1) components, you must first install the Oracle HTTP Server Release 12.2.1.1.0.

To install the Oracle HTTP Server:

1. Log in to the host where you want to install the Oracle HTTP Server.

2. Download the Oracle HTTP Server product distribution (fmw_12.2.1.1.0_ohs_linux64.bin | .exe) from [Oracle Technology Network](#) or [Oracle Software Delivery Cloud](#) on your host system.
3. Check that your machines meet the following requirements:
 - Ensure that the system, patch, kernel, and other requirements are met as specified in *Installing and Configuring Oracle HTTP Server*.
 - Because Oracle HTTP Server is installed by default on port 7777, you must ensure that port 7777 is not used by any service on the nodes. To check if this port is in use, run the following command before installing Oracle HTTP Server. You must free the port if it is in use.

```
netstat -an | grep 7777
```

4. On UNIX platforms, if the `/etc/oraInst.loc` file exists, check that its contents are correct. Specifically, check that the inventory directory is correct and that you have write permissions for that directory.

If the `/etc/oraInst.loc` file does not exist, you can skip this step.

5. The Oracle HTTP Server distribution is a .bin file. Run the installer by entering the following command:

```
./distribution_filename.bin
```

6. On the Installation Inventory Setup screen, specify the location where you want to create your central inventory in the **Inventory Directory** field.

Make sure that the Operating System Group name that you select on this screen has write permissions to the central inventory location. Click **OK**.

Note: This screen does not appear on Windows operating systems. For more information about the central inventory, see *Understanding the Oracle Central Inventory in Installing Software with the Oracle Universal Installer*.

7. On the Welcome screen, review the information and click **Next**.
8. On the Auto Updates screen, select one of the following 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.
9. On the Installation Location screen, specify the location for the Oracle home directory and click **Next**.

Since you are installing the standard installation topology for a collocated Oracle HTTP Server in a WebLogic Server domain, enter the path to an existing Oracle Fusion Middleware Infrastructure Oracle home.

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

10. On the Installation Type screen, select **Collocated HTTP Server (Managed through WebLogic Server)** and click **Next**.

For more information about the installation types, see *Introducing the Oracle HTTP Server Standard Installation Topologies* in *Installing and Configuring Oracle HTTP Server*.

11. On the Prerequisite Check screen, verify that your system meets the minimum necessary requirements and click **Next**. If you see a warning or error message, see *Oracle Fusion Middleware System Requirements and Specifications*.

For more information about other options on this screen, click **Help**.

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%.

If the installation is successful, the Installation Complete screen displays the message *Oracle HTTP Server (OHS) installation completed successfully*. Click **Finish** to dismiss the installer.

You have installed the Oracle HTTP Server in a collocated mode.

8.2 Upgrading the Existing 12c Schema with the Upgrade Assistant

You need to upgrade the existing 12c OPSS (Oracle Platform Security Services) and IAU (Audit Services) schema using the 12c Upgrade Assistant.

To upgrade the OPSS and IAU 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 Selected Schemas screen, select **Individually Selected Schemas** and click **Next**.

The Upgrade Assistant identifies the components that are available for a schema upgrade thus allowing you to select the schemas you want to include in the upgrade.

Caution: Upgrade only those schemas that are used to support your 12.2.1.0 components. Do not upgrade schemas that are currently being used to support 11g or 12c components that are not included in the Oracle Fusion Middleware 12.2.1.0 release.

4. The Available Components screen provides a list of installed Oracle Fusion Middleware components that have schemas that can be upgraded. When you select a component, the schemas and any dependencies are automatically selected.

Select **Oracle Platform Security Services**. Selecting this automatically selects **Oracle Audit Services**. Click **Next**.

5. On the Domain Directory screen, enter the 12c (12.1.2, 12.1.3, or 12.2.1.0) WebLogic domain directory. Click **Browse** and use the navigation tree to select the 12c WebLogic domain directory.

The Upgrade Assistant requires the 12c domain location to access the jps-config.xml file.

6. 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.

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.

8.3 Reconfiguring the Existing 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.

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 8-1 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.

Table 8-1 (Cont.) Field Descriptions for Node Manager Screen

Option	Description
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 ListenAddress, ListenPort, StartScriptName, JavaHome, and LogFile.
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.
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.

8.4 Upgrading the Component Configuration with the Upgrade Assistant

If you are running the Upgrade Assistant from an Oracle home that contains managed domain components, then the **WebLogic Component Configuration** upgrade option is available.

To upgrade the component configurations for Oracle HTTP Server:

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 WebLogic Components screen, select the **WebLogic Component Configurations** option to upgrade component configurations for a managed

WebLogic Server domain. Enter the connection details required to connect to the WebLogic Administration Server that is managing the domain and 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. 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.
-

7. 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.

8. 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.

9. 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.

10. 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.

8.5 Post-Upgrade Tasks

The post-upgrade tasks include starting the Node Manager, starting the Administration Server, starting the Oracle HTTP Server, and verifying whether the upgrade from previous 12c release to this release is successful.

Starting the Node Manager

Server instances in a WebLogic Server production environment are often distributed across multiple domains, machines, and geographic locations. Node Manager is a WebLogic Server utility that enables you to start, shut down, and restart Administration Server and Managed Server instances from a remote location.

Starting the Administration Server

One instance of WebLogic Server in each domain acts as an 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 Oracle HTTP Server

You can use Enterprise Manager Fusion Middleware Control or the `opmnctl` command to start, stop, and restart Oracle HTTP Server.

Verifying that Oracle HTTP Server Installation is Successful

If you can log in to the Administration Console and the Enterprise Manager Fusion Middleware Control, you have installed Oracle HTTP Server successfully.

8.5.1 Starting the Node Manager

Server instances in a WebLogic Server production environment are often distributed across multiple domains, machines, and geographic locations. Node Manager is a WebLogic Server utility that enables you to start, shut down, and restart Administration Server and Managed Server instances from a remote location.

On Windows operating systems, it is recommended that you configure Node Manager to run as a startup service. This allows Node Manager to start up automatically each time the system is restarted. For more information, see *Running Node Manager as a Startup Service* in *Administering Node Manager for Oracle WebLogic Server*.

Note: On UNIX platform, do not run Node Manager as the root user.

To start the Node Manager:

1. Change directory to the following:

On UNIX operating system:

```
 $ORACLE_HOME/wlserver/server/bin/
```

On Windows operating system:

```
 %ORACLE_HOME%\wlserver\server\bin\
```

2. On the UNIX operating system, start the Node Manager by running the following command:

```
 nohup ./startNodeManager.sh > nm.out&
```

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

On the Windows operating system, start the Node Manager by running the following command:

```
 startNodeManager.cmd
```

You can also use a shortcut on the **Start** menu to start the Node Manager. Click **Tools** and then click **Node Manager**.

8.5.2 Starting the Administration Server

One instance of WebLogic Server in each domain acts as an 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.

In a development environment, it is usually sufficient to start an Administration Server and deploy your applications directly onto the Administration Server. However, in a production environment, it is recommended that you create Managed Servers to run your applications.

To start an Administration Server:

1. Change directory to the following:

```
$DOMAIN_HOME/DOMAIN_NAME
```

Where, *DOMAIN_NAME* is the name of the directory in which you located the domain.

2. On the Windows operating system, start the Administration Server by running the following command:

```
startWebLogic.cmd
```

On the Windows operating system, the Configuration Wizard creates a shortcut on the Start menu to start the Administration Server that you created.

On the UNIX operating system, start the Administration Server by running the following command:

```
./startWebLogic.sh
```

Note: If the server prompts you to enter a username and password, enter the name of a WebLogic Server user who has permission to start servers.

8.5.3 Starting the Oracle HTTP Server

You can use Enterprise Manager Fusion Middleware Control or the `opmnctl` command to start, stop, and restart Oracle HTTP Server.

Starting the Oracle HTTP Server using the Enterprise Manager Fusion Middleware Control

To start an Oracle HTTP Server using the Enterprise Manager Fusion Middleware Control:

1. Go to the Oracle HTTP Server home page.
2. From the Oracle HTTP Server menu:
 - a. Select **Control**.
 - b. Select **Start Up** from the **Control** menu.Or
3. From the Target Navigation tree:
 - a. Right-click the Oracle HTTP Server instance you want to start.
 - b. Select **Control**.
 - c. Select **Start Up** from the **Control** menu.

Starting the Oracle HTTP Server using `opmnctl`

To start Oracle HTTP Server components in an Oracle instance using `opmnctl`:

1. Change directory to the following:

```
ORACLE_INSTANCE/bin
```

2. Run the following command to start all Oracle HTTP Server components in an Oracle instance

```
opmnctl startproc process-type=OHS
```

Determining the Status of Oracle HTTP Server Components using `opmnctl`

To determine the status of the Oracle HTTP Server components:

1. Change directory to the following:

```
ORACLE_INSTANCE/bin
```

2. Enter the following command:

```
opmnctl status
```

Following is a sample output:

```
Processes in Instance: instancel
```

ias-component	process-type	pid	status
webcache1	WebCache-admin	19556	Alive
webcache1	WebCache	19555	Alive
ohs1	OHS	7249	Alive

8.5.4 Verifying that Oracle HTTP Server Installation is Successful

If you can log in to the Administration Console and the Enterprise Manager Fusion Middleware Control, you have installed Oracle HTTP Server successfully.

To verify whether the software patch is installed successfully:

1. Log in to the following URL:

```
http://MachineName.us.oracle.com:7777
```

2. To access Enterprise Manager Fusion Middleware Control:

```
http://MachineName.us.oracle.com:7001/em
```

