

Oracle® Fusion Middleware

Upgrading with the Upgrade Assistant



12c (12.2.1.3)

E95498-02

July 2019

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

ORACLE®

Oracle Fusion Middleware Upgrading with the Upgrade Assistant, 12c (12.2.1.3)

E95498-02

Copyright © 2015, 2019, Oracle and/or its affiliates. All rights reserved.

Primary Author: Oracle Corporation

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

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

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

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs 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

Audience	v
Documentation Accessibility	v
Related Documents	vi
Conventions	vi

1 About Using the Oracle Fusion Middleware Upgrade Assistant

About the Upgrade Assistant	1-1
About Using the Upgrade Assistant in the Upgrade Process	1-2
About Using the Upgrade Assistant to Create Schemas	1-3
About Using the Upgrade Assistant to Perform a Readiness Check Before an Upgrade	1-3
Upgrading Product Schemas	1-4
About Using the Upgrade Assistant to Upgrade Component Configurations	1-4
Identifying Existing Schemas Available for Upgrade	1-4
Identifying Schemas that Can Be Upgraded to 12c (12.2.1.3.0)	1-5
Identifying Configurations That Can Be Upgraded with the Upgrade Assistant	1-8

2 Using the Upgrade Assistant to Perform an Upgrade

Before You Use the Upgrade Assistant	2-1
Creating a Complete Backup	2-2
Backing Up the Schema Version Registry Table	2-3
Maintaining Customized Domain and Environment Settings	2-3
Special Considerations for Online Backup and Recovery	2-4
Saving Grants on SYS Owned Objects	2-5
Exporting Schemas Before You Upgrade	2-5
Identifying Queue States Before an Upgrade	2-6
Checking for Invalid Database Objects Before an Upgrade	2-6
Creating a Non-SYSDBA User to Run the Upgrade Assistant	2-6
Stopping Servers and Processes	2-9
Starting the Upgrade Assistant	2-10

Upgrade Assistant Parameters	2-11
Running a Pre-Upgrade Readiness Check	2-12
About Running a Pre-Upgrade Readiness Check	2-13
Starting the Upgrade Assistant in Readiness Mode	2-13
Upgrade Assistant Parameters	2-14
Performing a Readiness Check with the Upgrade Assistant	2-16
Understanding the Readiness Report	2-17
Understanding the Product Schemas Upgrade	2-21
Understanding the Component Configurations Upgrade	2-26
Performing Post-Upgrade Procedures	2-28
Performing Basic Post-Upgrade Administrative Tasks	2-28
Verifying a Successful Schema Upgrade	2-30
Checking for Invalid Database Objects After an Upgrade	2-30

3 Performing a Silent Upgrade with a Response File

Creating an Upgrade Response File	3-1
Using a Response File to Upgrade Fusion Middleware	3-2
Upgrade Assistant Parameters	3-2

4 Troubleshooting Your Upgrade

Troubleshooting an Upgrade	4-1
Reviewing Log Files	4-3
Investigating Examination Failures	4-4
Investigating Upgrade Failures	4-4
Resolving Common Upgrade Assistant Errors	4-5
Ensuring there is sufficient disk space	4-5
Resolving Database Connection Problems When Upgrading Schemas	4-6
Attempting to Upgrade an Unsupported Domain	4-6
Restarting the Upgrade Assistant After a Failure	4-6

Preface

This guide describes how to use the Oracle Fusion Middleware Upgrade Assistant, including how to prepare for an upgrade, how the upgrade will affect the existing environment, and how to prepare your database and repository schemas for an upgrade.

- [Audience](#)
- [Documentation Accessibility](#)
Get more information about Oracle's commitment to accessibility and support through My Oracle Support.
- [Related Documents](#)
Upgrade documentation is organized by tasks in the 12c documentation library. The task-specific pages provide direct links to common upgrade procedures and related documentation.
- [Conventions](#)

Audience

This guide is intended for 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 12c system administration and configuration information for the existing deployment
- The configuration and expected behavior of the system or systems being upgraded

Documentation Accessibility

Get more information about Oracle's commitment to accessibility and support through My Oracle Support.

For information about Oracle's commitment to accessibility, see Oracle Accessibility Program website at <http://www.oracle.com>.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support.

For details, see [My Oracle Support](#). If you are hearing impaired, see <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>.

Related Documents

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

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

- For installation information, see Fusion Middleware Installation Documentation.
- For upgrade information, including upgrade videos, see Fusion Middleware 12c Upgrade Documentation.
- For administration-related information, see Fusion Middleware 12c Administration Documentation.
- For release-related information, see Fusion Middleware 12c Release Notes.

Conventions

The following text conventions are used in this document:

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

1

About Using the Oracle Fusion Middleware Upgrade Assistant

Use the Upgrade Assistant to upgrade and configure supported Fusion Middleware 11g and 12c domains to 12c (12.2.1.3.0).

The Upgrade Assistant automates many common upgrade tasks including schema upgrades and domain reconfigurations. You can also use the Upgrade Assistant to perform a readiness check prior to an upgrade.

Note:

This guide is meant to be used as a reference tool for your upgrade. Always consult your component-specific upgrade documentation for detailed information about the supported upgrade starting points, as well as the prerequisites and upgrade paths for specific installation types.

- [About the Upgrade Assistant](#)
Use the Upgrade Assistant to upgrade supported component schemas, component configuration data, and standalone system component configurations in 11g and previous 12c releases to 12c (12.2.1.3.0). The Upgrade Assistant can also be used to run a pre-upgrade readiness check.
- [About Using the Upgrade Assistant in the Upgrade Process](#)
Use the Upgrade Assistant in conjunction with other Oracle Fusion Middleware tools and processes to upgrade supported Fusion Middleware 11g and 12c components to the latest release.
- [Identifying Existing Schemas Available for Upgrade](#)
This optional task enables you to review the list of available schemas before you begin the upgrade by querying the schema version registry. The registry contains schema information such as version number, component name and ID, date of creation and modification, and custom prefix.
- [Identifying Schemas that Can Be Upgraded to 12c \(12.2.1.3.0\)](#)
Before an upgrade, it is important to know which schemas can be upgraded to 12c (12.2.1.3.0). Not all schemas require an upgrade, and not all schemas can be upgraded.
- [Identifying Configurations That Can Be Upgraded with the Upgrade Assistant](#)
After you have reconfigured the domain using the Reconfiguration Wizard, use the Upgrade Assistant to upgrade and configure the supported component configurations to 12c (12.2.1.3.0).

About the Upgrade Assistant

Use the Upgrade Assistant to upgrade supported component schemas, component configuration data, and standalone system component configurations in 11g and previ-

ous 12c releases to 12c (12.2.1.3.0). The Upgrade Assistant can also be used to run a pre-upgrade readiness check.

The Upgrade Assistant is available at the following location in your Fusion Middleware installation:

(UNIX) `ORACLE_HOME/oracle_common/upgrade/bin`

(Windows) `ORACLE_HOME\oracle_common\upgrade\bin`

When you run the Upgrade Assistant, it performs the following tasks:

- When run in `readiness` mode, the Upgrade Assistant runs a pre-upgrade check on the schemas and component configurations associated with a domain.
- For schema upgrades, the Administration server is offline and the list of schemas to upgrade is determined by reading the domain configuration.
- For component configuration upgrades, the Administration server is offline and the list of component configurations to upgrade is determined by reading the domain configuration.

 **Note:**

Before you start the Upgrade Assistant, Oracle recommends that you create a non-SYSDBA user. See [Creating a Non-SYSDBA User to Run the Upgrade Assistant](#).

About Using the Upgrade Assistant in the Upgrade Process

Use the Upgrade Assistant in conjunction with other Oracle Fusion Middleware tools and processes to upgrade supported Fusion Middleware 11g and 12c components to the latest release.

The 11g to 12c upgrade process is different from previous releases. To understand how the Upgrade Assistant and the other upgrade tools are used, see [Understanding the 12c Upgrade Process in *Planning an Upgrade of Oracle Fusion Middleware*](#)

In a typical upgrade, the Upgrade Assistant may be used several times.

- [About Using the Upgrade Assistant to Create Schemas](#)
The Upgrade Assistant calls Repository Creation Utility (RCU) APIs internally to create missing schemas which are required for 12c. You do not have to run the RCU separately unless you want to customize the default schema settings.
- [About Using the Upgrade Assistant to Perform a Readiness Check Before an Upgrade](#)
Run the readiness check before you start an actual upgrade to determine if there are any issues that could prevent a successful upgrade.
- [Upgrading Product Schemas](#)
After stopping servers and processes, use the Upgrade Assistant to upgrade supported product schemas to the current release of Oracle Fusion Middleware.
- [About Using the Upgrade Assistant to Upgrade Component Configurations](#)
After the domain has been reconfigured, use the Upgrade Assistant to upgrade any component configurations.

About Using the Upgrade Assistant to Create Schemas

The Upgrade Assistant calls Repository Creation Utility (RCU) APIs internally to create missing schemas which are required for 12c. You do not have to run the RCU separately unless you want to customize the default schema settings.

The Upgrade Assistant can now detect when component schemas may be missing and must be created before the upgrade. If enabled, the Upgrade Assistant can create these schemas for you using the default tablespace and temporary tablespace settings. To customize the schema settings you must use the Repository Creation Utility to create the schemas.

 **Note:**

If you do not want the Upgrade Assistant to create these schemas for you, you must deselect the **Create missing schemas for the specified domain** option.

About Using the Upgrade Assistant to Perform a Readiness Check Before an Upgrade

Run the readiness check before you start an actual upgrade to determine if there are any issues that could prevent a successful upgrade.

The Upgrade Assistant can be run in `-readiness` mode to identify potential upgrade issues before you perform an actual upgrade. You can perform a readiness check while the system is online. It is a read-only operation that scans your existing domain or database schemas and produces a text file with the results of the scan. A readiness check does not require that schemas are part of any particular domain. If your pre-upgrade environment has issues, you can correct those issues and then rerun a readiness check before you upgrade.

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

Understanding the Difference between the Examine Phase and a Readiness Check

The Examine phase of the upgrade process is a separate operation from running a readiness check. Typically, readiness checks are more thorough than the checks performed during the Examine phase. A readiness check is a read-only process performed before you start the actual upgrade, while the Examine phase occurs just before you click **Upgrade** and identifies potential issues with the schemas and configuration data.

 **Note:**

If you have performed any of the upgrade steps, do not run a readiness check, or the results may be invalid.

 **Caution:**

If you encounter issues that prevent a successful upgrade, you may have to revert to your backed up, pre-upgrade environment and start the upgrade again. Running a readiness check can help identify some of these issues and prevent extended system downtime.

Upgrading Product Schemas

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

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

About Using the Upgrade Assistant to Upgrade Component Configurations

After the domain has been reconfigured, use the Upgrade Assistant to upgrade any component configurations.

During the component configuration process, the domain's configuration files are updated to get them in-sync with the target version. As part of this process, the existing configuration data, which has been rendered invalid by the newer release, is deleted and replaced with configuration data that is compatible with the newer version.

Identifying Existing Schemas Available for Upgrade

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

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

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

```
SET LINE 120
COLUMN MRC_NAME FORMAT A14
COLUMN COMP_ID FORMAT A20
```

```
COLUMN VERSION FORMAT A12
COLUMN STATUS FORMAT A9
COLUMN UPGRADED FORMAT A8
SELECT MRC_NAME, COMP_ID, OWNER, VERSION, STATUS, UPGRADED FROM SCHE-
MA_VERSION_REGISTRY ORDER BY MRC_NAME, COMP_ID;
```

2. Examine the report that is generated.

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

3. Note the schema prefix name that was used for your existing schemas. You will use the same prefix when you create new 12c (12.2.1.3.0) schemas.

 **Notes:**

- If your existing schemas are not from a supported version, then you must upgrade them to a supported version before using the 12c (12.2.1.3.0) upgrade procedures. Refer to your pre-upgrade version documentation for more information.
- Some components, such as Oracle Enterprise Data Quality, Oracle GoldenGate Monitor, and Oracle GoldenGate Veridata, support an upgrade from versions other than the standard Oracle Fusion Middleware supported versions.
- If you used an OID-based policy store in the earlier versions, make sure to create a new OPSS schema before you perform the upgrade. After the upgrade, the OPSS schema remains an LDAP-based store.
- You can only upgrade schemas for products that are available for upgrade in Oracle Fusion Middleware release 12c (12.2.1.3.0). Do not attempt to upgrade a domain that includes components that are not yet available for upgrade to 12c (12.2.1.3.0).

Identifying Schemas that Can Be Upgraded to 12c (12.2.1.3.0)

Before an upgrade, it is important to know which schemas can be upgraded to 12c (12.2.1.3.0). Not all schemas require an upgrade, and not all schemas can be upgraded.

When schemas are created in your database, the Repository Creation Utility (RCU) creates and maintains a table called `schema_version_registry`. Refer to this table before you perform an upgrade. The Schema Version After Upgrade column shows the latest version for that schema. If your schemas are already at this version, then you will not need to upgrade.

NOTE: The Schema(s) column shows the default schema name format of prefix and schema name separated by an underscore (`_`) character. The default prefix is `DEV`, but you may have created new prefixes for your schemas while using the RCU.

Table 1-1 Schemas that can be upgraded to Oracle Fusion Middleware 12c (12.2.1.3.0)

Component Name	Schemas	Schema Version Before Upgrade	Schema Version After Upgrade	Dependencies and Additional Information
Audit Services ¹	<i>prefix_IAU</i>	11.1.1.7	12.2.1.3	No dependencies.
		11.1.1.9		
		12.1.2.0		
		12.1.3.0		
		12.2.1.0		
		12.2.1.1		
		12.2.1.2		
Metadata Services	<i>prefix_MDS</i>	11.1.1.7	12.2.1.3	No dependencies.
		11.1.1.9		
		12.1.2.0		
		12.1.3.0		
		12.2.1.0		
		12.2.1.1		
		12.2.1.2		
Platform Security Services ²	<i>prefix_OPSS</i>	11.1.1.7	12.2.1.0	The <i>prefix_IAU</i> schema must be upgraded first. 12.2.1.0 is the latest version for this schema.
		11.1.1.9		
		12.1.2.0		
		12.1.3.0		
Common Infrastructure Services (LocalSvcTb)	<i>prefix_STB</i>	12.1.2.0	12.2.1.3	The <i>prefix_STB</i> schema must be upgraded when upgrading other components, such as SOAINFRA.
		12.1.3.0		
		12.2.1.0		
		12.2.1.1		
		12.2.1.2		
User Messaging Service	<i>prefix_ORASDPM</i> <i>prefix_UMS</i> ³	11.1.1.7	12.2.1.0	12.2.1.0 is the latest version for this schema.
		11.1.1.9		
		12.1.2.0		
		12.1.3.0		
WebLogic Server	<i>prefix_WLS</i>	11.1.1.7	12.2.1.0	12.2.1.0 is the latest version for this schema.
		12.1.2.0		
Data Integrator (Master and Work Repository)	<i>prefix_ODI_REPO</i>	11.1.1.7	12.2.1.3	No dependencies.
		11.1.1.9		
		12.1.2.0		
		12.1.3.0		
		12.2.1.0		
		12.2.1.1		
12.2.1.2				

Table 1-1 (Cont.) Schemas that can be upgraded to Oracle Fusion Middleware 12c (12.2.1.3.0)

Component Name	Schemas	Schema Version Before Upgrade	Schema Version After Upgrade	Dependencies and Additional Information
SOA Infrastructure (including Oracle Business Activity Monitoring, Business Process Management, and Oracle Service Bus)	<i>prefix_SOAINFRA</i>	11.1.1.7	12.2.1.3	The <i>prefix_STB</i> schema must be upgraded first.
		11.1.1.9		
		12.1.3		
		12.2.1.0		
		12.2.1.1		
Oracle Enterprise Scheduler	<i>prefix_ESS</i>	11.1.1.7	12.2.1.3	No dependencies.
		11.1.1.9		
		12.1.3		
		12.2.1.0		
		12.2.1.1		
Oracle Managed File Transfer	<i>prefix_MFT</i>	12.1.3.0	12.2.1.3	The <i>prefix_MDS</i> , <i>prefix_IAU</i> , and <i>prefix_UMS</i> must also be upgraded.
		12.2.1.0		
		12.2.1.1		
		12.2.1.2		
Oracle WebCenter Content Server	<i>prefix_OCS</i> <i>prefix_OCSSEARCH</i>	11.1.1.7	12.2.1.0	The <i>prefix_MDS</i> schema must be upgraded first. 12.2.1.0 is the latest version for this schema.
		11.1.1.8 ⁴		
		11.1.1.9		
Oracle WebCenter Content: Imaging	<i>prefix_IPM</i>	11.1.1.2.1	12.2.1.3	
Oracle WebCenter Content: Records	<i>prefix_URM</i>	11.1.1.7.0	12.2.1.0	12.2.1.0 is the latest version for this schema.
		11.1.1.8.0		
		11.1.1.9.0		
Oracle WebCenter Enterprise Capture	<i>prefix_CAPTURE</i>	11.1.1.8	12.2.1.0	The <i>prefix_MDS</i> and <i>prefix_OPSS</i> schemas must be upgraded first. 12.2.1.0 is the latest version for this schema.
Oracle WebCenter Portal	<i>prefix_PORTLET</i> <i>prefix_ACTIVITIES</i> <i>prefix_DISCUSSIONS</i> <i>prefix_DISCUSSIONS_CRAWLER</i>	11.1.1.7 ⁵	12.2.1.0	The <i>prefix_MDS</i> schema must be upgraded first. 12.2.1.0 is the latest version for this schema.
		11.1.1.8		

Table 1-1 (Cont.) Schemas that can be upgraded to Oracle Fusion Middleware 12c (12.2.1.3.0)

Component Name	Schemas	Schema Version Before Upgrade	Schema Version After Upgrade	Dependencies and Additional Information
Oracle WebCenter Portal	<i>prefix_WEBCENTER</i>	11.1.1.7 11.1.1.8 12.2.1.0	12.2.1.1	12.2.1.1 is the latest version for this schema.
Oracle WebCenter Sites	<i>prefix_WCSITES</i>	12.2.1.0 12.2.1.1 12.2.1.2	12.2.1.3	
Enterprise Data Quality	<i>prefix_EDQCONFIG</i> <i>prefix_EDQRE-SULTS</i> <i>prefix_EDQSTAGING</i>	11.1.1.7.3 12.1.3.0 12.2.1.0 12.2.1.1 12.2.1.2	12.2.1.3	
GoldenGate Monitor	<i>prefix_OGGMON</i>	11.2.1.0.7 12.1.3.0	12.2.1.0	12.2.1.0 is the latest version for this schema.
GoldenGate Studio	<i>prefix_OGGSTUDIO_REPO</i>	12.2.1.0	12.2.1.1	12.2.1.1 is the latest version for this schema.
GoldenGate Veridata	<i>prefix_VERIDATA</i>	11.2.1.0.1 12.1.3.0 12.2.1.0	12.2.1.2	

- ¹ When upgrading *_IAU* from 11g starting points, the Upgrade Assistant updates two auxiliary schemas, *IAU_APPEND* and *IAU_VIEWER* and adds them to the *schema_version_registry* for 12c (12.2.1.3.0). These schemas were not part of the 11g *schema_version_registry* table.
- ² As of release 11.1.1.7, OPSS audit data is stored in IAU common tables instead of the JPS table.
- ³ When upgrading a system from a previous 12c release, the schema is called *prefix_UMS*.
- ⁴ First release to support Edition-Based Redefinition (EBR) enabled schemas.
- ⁵ First release to support Edition-Based Redefinition (EBR) enabled schemas.

Identifying Configurations That Can Be Upgraded with the Upgrade Assistant

After you have reconfigured the domain using the Reconfiguration Wizard, use the Upgrade Assistant to upgrade and configure the supported component configurations to 12c (12.2.1.3.0).

When you use the Upgrade Assistant to upgrade your configurations, it automatically determines the components that are configured and can be upgraded to the latest version. Before you begin the upgrade, review the list of components that can be upgraded to this release.

 **Caution:**

Do not attempt an upgrade if you want to continue using Fusion Middleware 11g components that have been deprecated or are not yet available for use in the 12c (12.2.1.3.0) release.

Component configurations that can be upgraded include:

- Oracle Internet Directory
- Common Infrastructure Services
- Oracle Data Integrator
- Oracle Enterprise Data Quality
- Oracle Forms
- Oracle GoldenGate Monitor
- Oracle GoldenGate Veridata
- Oracle HTTP Server
- Oracle MapViewer
- Oracle Reports
- Oracle Traffic Director
- Oracle User Messaging Service
- Oracle WebCenter Portal
- Oracle WebCenter Sites
- Oracle Web Services Manager
- System Components Infrastructure
- Java Required Files (JRF) Infrastructure

2

Using the Upgrade Assistant to Perform an Upgrade

The Upgrade Assistant is used in different ways to upgrade schemas, component configurations, and to perform a readiness check on your pre-upgrade environment.

 **Note:**

This chapter provides a high-level overview of how to use the Upgrade Assistant to perform an Oracle Fusion Middleware upgrade. Use the component-specific upgrade guides when performing an actual upgrade.

- [Before You Use the Upgrade Assistant](#)
Before beginning an upgrade with the Upgrade Assistant, you should create a complete backup and perform other pre-upgrade checks and tasks.
- [Starting the Upgrade Assistant](#)
Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12c (12.2.1.3.0). Oracle recommends that you run the Upgrade Assistant as a non-SYSDBA user, completing the upgrade for one domain at a time.
- [Running a Pre-Upgrade Readiness Check](#)
To identify potential issues with the upgrade, Oracle recommends that you run a readiness check before you start the upgrade process. Be aware that the readiness check may not be able to discover all potential issues with your upgrade. An upgrade may still fail, even if the readiness check reports success.
- [Understanding the Product Schemas Upgrade](#)
Navigate through the screens in the Upgrade Assistant to upgrade the product schemas.
- [Understanding the Component Configurations Upgrade](#)
Navigate through the screens in the Upgrade Assistant to upgrade component configurations in the WebLogic domain.
- [Performing Post-Upgrade Procedures](#)
Complete any additional post-upgrade configuration tasks after the upgrade to verify that your newly upgrade domain is functioning as expected. Perform only those tasks that apply to your domain configuration.

Before You Use the Upgrade Assistant

Before beginning an upgrade with the Upgrade Assistant, you should create a complete backup and perform other pre-upgrade checks and tasks.

 **Note:**

You may be required to perform additional tasks before starting your actual upgrade process. Each of the component-specific upgrade guides provides a checklist that includes a complete list of a pre-upgrade tasks that must be performed before starting an upgrade.

- [Creating a Complete Backup](#)
Before you start an upgrade, back up all system-critical files, including the databases that host your Oracle Fusion Middleware schemas.
- [Special Considerations for Online Backup and Recovery](#)
Perform these additional backup tasks if your environment includes multiple middleware homes, and performing a full database restore after an upgrade failure is not a desirable option.
- [Checking for Invalid Database Objects Before an Upgrade](#)
To identify invalid objects that may cause the upgrade to fail, recompile database objects before you run the Upgrade Assistant.
- [Creating a Non-SYSDBA User to Run the Upgrade Assistant](#)
Oracle recommends that you create a non-SYSDBA user called `FMW` to run the Upgrade Assistant. This user has the privileges required to modify schemas, but does not have full administrator privileges.
- [Stopping Servers and Processes](#)
Before you run the Upgrade Assistant to upgrade your schemas and configurations, you must shut down all of the pre-upgrade processes and servers, including the Administration Server and any managed servers.

Creating a Complete Backup

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

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

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

See:

- [Backing Up Your Environment in *Administering Oracle Fusion Middleware*](#)
- [Upgrading and Preparing Your Oracle Databases for 12c in *Planning an Upgrade of Oracle Fusion Middleware*](#)
- [Backing Up the Schema Version Registry Table](#)
Your system backup must include the `SYSTEM.SCHEMA_VERSION_REGISTRY$` table or the `FMWREGISTRY.SCHEMA_VERSION_REGISTRY$` table.
- [Maintaining Customized Domain and Environment Settings](#)
If you have modified any domain-generated, server startup scripts, or configuration files in your pre-upgrade environment, it is important to note that these changes are overwritten during the installation, domain upgrade, and reconfiguration opera-

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

Backing Up the Schema Version Registry Table

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

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

Note:

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

Maintaining Customized Domain and Environment Settings

If you have modified any domain-generated, server startup scripts, or configuration files in your pre-upgrade environment, it is important to note that these changes are overwritten during the installation, domain upgrade, and reconfiguration operations. Save your customized files to a shared library location so that you can continue to use them after the upgrade.

Every domain installation includes dynamically-generated domain and server startup scripts, such as `setDomainEnv`. These files are replaced by newer versions during the installation and upgrade process. To maintain your custom domain-level environment settings, Oracle recommends that you create a separate file to store the custom domain information before you upgrade, instead of modifying the scripts directly.

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

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

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

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

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

 **Note:**

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

Special Considerations for Online Backup and Recovery

Perform these additional backup tasks if your environment includes multiple middle-ware homes, and performing a full database restore after an upgrade failure is not a desirable option.

Understanding the Impact of a Full Database Restore

It is important that you understand the impact of a full database restore when creating your backup and recovery plan. If your upgrade fails, you may be required to perform a complete database restore. However, in some cases this may not be possible or desirable.

- Is your database shared by production environments that must remain online when a single FMW home is being upgraded?
- Does your database need to remain online when recovering from a failed upgrade?
- Is performing a full database restore an undesirable solution for recovering from a failed upgrade?

If you answered 'yes' to any of the following questions, then complete these additional pre-upgrade tasks before you begin:

- [Saving Grants on SYS Owned Objects](#)
In the event of an upgrade failure, all grants to SYS owned objects will be lost when the schema is dropped. Oracle recommends that you create a script that can be used to re-apply the grants if necessary.
- [Exporting Schemas Before You Upgrade](#)
Use data pump export to backup the schemas that will be upgraded.
- [Identifying Queue States Before an Upgrade](#)
In the event of a an upgrade failure, the queues must be manually restarted. Take inventory of these queues to assist in restarting them.

Saving Grants on SYS Owned Objects

In the event of an upgrade failure, all grants to SYS owned objects will be lost when the schema is dropped. Oracle recommends that you create a script that can be used to re-apply the grants if necessary.

An example of how to create this script is shown below. Please note the following about the generated SQL script:

- The spooled output will need to be edited before it can be executed by SQLPLUS, the text of the SQL queries and the "spool off" command need to be removed from the spooled file.
- Some of the grants may return errors when being applied after a drop/import of a schema. Some instances where this is not a fatal error are:
 - The grant already exists
 - The name of the grant object is dynamically generated when the schema is created. For example, advanced queueing views are named QTnnnnnnnn_BUFFER.

Sample SQLPlus commands to create a script for re-applying grants:

```
# The schema prefix in this example is "DEV"
$ORACLE_HOME/bin/sqlplus username/password
exec dbms_metadata.set_transform_param(dbms_metadata.SESSION_TRANSFORM, 'SQLTERMINATOR', TRUE);
set long 100000
set longchunksize 100000
set lines 1000
set termout off echo off newp 0 spa 0 pages 0 feed off head off trims on
tab off
spool /tmp/create-grants.sql
select dbms_metadata.get_granted_ddl ('OBJECT_GRANT',username) from
all_users where username in ('DEV_MDS', 'DEV_IAU', 'DEV_IAU_APPEND',
'DEV_IAU_VIEWER', 'DEV_OPSS', 'DEV_UMS', 'DEV_WLS', 'DEV_SOAINFRA',
'DEV_STB', 'DEV_ESS')
union all
select dbms_metadata.get_granted_ddl ('SYSTEM_GRANT',username) from
all_users where username in ('DEV_MDS', 'DEV_IAU', 'DEV_IAU_APPEND',
'DEV_IAU_VIEWER', 'DEV_OPSS', 'DEV_UMS', 'DEV_WLS', 'DEV_SOAINFRA',
'DEV_STB', 'DEV_ESS')
union all
select dbms_metadata.get_granted_ddl ('DEFAULT_ROLE',username) from
all_users where username in ('DEV_MDS', 'DEV_IAU', 'DEV_IAU_APPEND',
'DEV_IAU_VIEWER', 'DEV_OPSS', 'DEV_UMS', 'DEV_WLS', 'DEV_SOAINFRA',
'DEV_STB', 'DEV_ESS');
spool off
```

Exporting Schemas Before You Upgrade

Use data pump export to backup the schemas that will be upgraded.

Refer to the [Oracle Database Utilities](#) guide for information on using Oracle Data Pump.

The following example shows a sample export:

```
# The schema prefix in this example is "DEV"
# The schemas being exported are for the SOA, BPM and ESS environments
$ORACLE_HOME/bin/sqlplus username/password
create directory data_pump_directory as '/scratch/db12cr2/export';

expdp username/password schemas=DEV_STB,DEV_SOAINFRA,DEV_IAU_VIEW-
ER,DEV_MDS,DEV_IAU_APPEND,DEV_WLS,DEV_UMS,DEV_OPSS,DEV_IAU,DEV_ESS directo-
ry=data_pump_directory dumpfile=export.dmp compression=ALL
```

Identifying Queue States Before an Upgrade

In the event of a an upgrade failure, the queues must be manually restarted. Take inventory of these queues to assist in restarting them.

The restoration of a single schema will not restart any queues that are imported. You will need to restart all of the enabled queues. The following example shows the SQL commands that can be used to generate a list of the queues that would need to be restarted in the event of a failed upgrade. Provide the correct schema prefix for each schema owner.

```
set pagesize 20;
set linesize 200;
COLUMN OWNER FORMAT A50
COLUMN NAME FORMAT A50
select owner,name,enqueue_enabled,dequeue_enabled from dba_queues where
owner='DEV_SOAINFRA' ;
```

Checking for Invalid Database Objects Before an Upgrade

To identify invalid objects that may cause the upgrade to fail, recompile database objects before you run the Upgrade Assistant.

If you are using an Oracle database, you can recompile database objects before running the Upgrade Assistant by connecting to the database as SYS and running the following command from SQL*Plus to compile the database objects:

```
SQL> @oracle_home/software/rdbms/admin/utlrp.sql
```

Then, use the following query to ensure there are no invalid database objects:

```
SELECT owner, object_name FROM all_objects WHERE status='INVALID';
```

If there are invalid objects, run the `utlrp.sql` command again. If the problem persists, you should file a service request.

Creating a Non-SYSDBA User to Run the Upgrade Assistant

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

SYSDBA is an administrative privilege that is required to perform high-level administrative operations such as creating, starting up, shutting down, backing up, or recover-

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

 **Notes:**

The non-SYSDBA user FMW is created solely for the purpose of running the Upgrade Assistant. After this step is complete, drop the FMW user. Note that privileges required for running the Upgrade Assistant may change from release to release.

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

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

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

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

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

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

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

Stopping Servers and Processes

Before you run the Upgrade Assistant to upgrade your schemas and configurations, you must shut down all of the pre-upgrade processes and servers, including the Administration Server and any managed servers.

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

 **Note:**

The procedures in this section describe how to stop the existing, pre-upgrade servers and processes using the WLST command-line utility or a script. You can also use the Oracle Fusion Middleware Control and the Oracle WebLogic Server Administration Console. See Starting and Stopping Administration and Managed Servers and Node Manager.

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

Step 1: Stop System Components

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

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

You can stop system components in any order.

Step 2: Stop the Managed Servers

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

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

When prompted, enter your user name and password.

Step 3: Stop Oracle Identity Management Components

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

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

Step 4: Stop the Administration Server

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

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

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

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

Step 5: Stop Node Manager

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

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

Starting the Upgrade Assistant

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

To start the Upgrade Assistant:

Note:

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

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

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

- [Upgrade Assistant Parameters](#)

Upgrade Assistant Parameters

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

Table 2-1 Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
-readiness	Required for readiness checks Note: Readiness checks cannot be performed on standalone installations (those not managed by the WebLogic Server).	Performs the upgrade readiness check without performing an actual upgrade. Schemas and configurations are checked. Do not use this parameter if you have specified the <code>-examine</code> parameter.
-threads	Optional	Identifies the number of threads available for concurrent schema upgrades or readiness checks of the schemas. The value must be a positive integer in the range 1 to 8. The default is 4.
-response	Required for silent upgrades or silent readiness checks	Runs the Upgrade Assistant using inputs saved to a response file generated from the data that is entered when the Upgrade Assistant is run in GUI mode. Using this parameter runs the Upgrade Assistant in <i>silent mode</i> (without displaying Upgrade Assistant screens).
-examine	Optional	Performs the examine phase but does not perform an actual upgrade. Do not specify this parameter if you have specified the <code>-readiness</code> parameter.
-logLevel <i>attribute</i>	Optional	Sets the logging level, specifying one of the following attributes: <ul style="list-style-type: none"> TRACE NOTIFICATION WARNING ERROR INCIDENT_ERROR The default logging level is NOTIFICATION. Consider setting the <code>-logLevel TRACE</code> attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if <code>-logLevel TRACE</code> is used.

Table 2-1 (Cont.) Upgrade Assistant Command-Line Parameters

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

Running a Pre-Upgrade Readiness Check

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

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

About Running a Pre-Upgrade Readiness Check

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

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

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

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

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

 **Note:**

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

Starting the Upgrade Assistant in Readiness Mode

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

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

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

 **Note:**

If the `DISPLAY` environment variable is not set up properly to allow for GUI mode, you may encounter the following error:

```
Xlib: connection to ":1.0" refused by server
Xlib: No protocol specified
```

To resolve this issue, set the `DISPLAY` environment variable to the system name or IP address of your local workstation, and rerun Upgrade Assistant.

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

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

- [Upgrade Assistant Parameters](#)

Upgrade Assistant Parameters

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

Table 2-2 Upgrade Assistant Command-Line Parameters

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

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

Parameter	Required or Optional	Description
-examine	Optional	Performs the examine phase but does not perform an actual upgrade. Do not specify this parameter if you have specified the <code>-readiness</code> parameter.
-logLevel <i>attribute</i>	Optional	<p>Sets the logging level, specifying one of the following attributes:</p> <ul style="list-style-type: none"> • TRACE • NOTIFICATION • WARNING • ERROR • INCIDENT_ERROR <p>The default logging level is NOTIFICATION.</p> <p>Consider setting the <code>-logLevel TRACE</code> attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if <code>-logLevel TRACE</code> is used.</p>
-logDir <i>location</i>	Optional	<p>Sets the default location of upgrade log files and temporary files. You must specify an existing, writable directory where the Upgrade Assistant creates log files and temporary files. The default locations are:</p> <p>(UNIX)</p> <pre>NEW_ORACLE_HOME/ oracle_common/upgrade/ logs NEW_ORACLE_HOME/ oracle_common/upgrade/ temp</pre> <p>(Windows)</p> <pre>NEW_ORACLE_HOME\ora- cle_common\upgrade\logs NEW_ORACLE_HOME\ora- cle_common\upgrade\temp</pre>
-help	Optional	Displays all of the command-line options.

Performing a Readiness Check with the Upgrade Assistant

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

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

To complete the readiness check:

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

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

- **Include checks for all schemas** to discover and review all components that have a schema available to upgrade.
- **Include checks for all configurations** to review component configurations for a managed WebLogic Server domain.

Click **Next**.

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

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

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

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

- Specify the domain directory.
- Specify schema credentials to connect to the selected schema: **Database Type**, **DBA User Name**, and **DBA Password**. Then click **Connect**.

 **Note:**

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

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

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

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

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

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

Click **Next**.

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

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

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

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

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

Understanding the Readiness Report

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

The format of the readiness report file is:

```
readiness_timestamp.txt
```

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

A readiness report contains the following information:

Table 2-3 Readiness Report Elements

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

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

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

```
This readiness check report was created on Tue May 30 11:15:52 EDT 2016
```

Log file is located at: *NEW_ORACLE_HOME*/oracle_common/upgrade/logs/
ua2016-05-30-11-14-06AM.log
Readiness Check Report File: *NEW_ORACLE_HOME*/oracle_common/upgrade/logs/
readiness2016-05-30-11-15-52AM.txt

Starting readiness check of components.

Oracle Metadata Services

Starting readiness check of Oracle Metadata Services.

Schema User Name: DEV11_MDS

Database Type: Oracle Database

Database Connect String: machinename@yourcompany.com

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

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

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

Starting schema test: TEST_REQUIRED_PROCEDURES Test that the schema contains all the required stored procedures

**EXCEPTION Schema is missing a required procedure: GETREPOSITORY-
FEATURES**

Completed schema test: TEST_REQUIRED_PROCEDURES --> Test that the schema contains all the required stored procedures +++ FAIL

Starting schema test: TEST_REQUIRED_VIEWS Test that the schema contains all the required database views

Completed schema test: TEST_REQUIRED_VIEWS --> Test that the schema contains all the required database views +++ PASS

Starting index test for table MDS_ATTRIBUTES: TEST_REQUIRED_INDEXES --> Test that the table contains all the required indexes

Completed index test for table MDS_ATTRIBUTES: TEST_REQUIRED_INDEXES --> Test that the table contains all the required indexes +++ PASS

Starting index test for table MDS_COMPONENTS: TEST_REQUIRED_INDEXES --> Test that the table contains all the required indexes

Completed index test for table MDS_TXN_LOCKS: TEST_REQUIRED_INDEXES --> Test that the table contains all the required indexes +++ PASS

Starting schema test: TEST_REQUIRED_TRIGGERS Test that the schema has all the required triggers

Completed schema test: TEST_REQUIRED_TRIGGERS --> Test that the schema has all the required triggers +++ PASS

Starting schema test: TEST_MISSING_COLUMNS Test that tables and views are not missing any required columns

Completed schema test: TEST_MISSING_COLUMNS --> Test that tables and views are not missing any required columns +++ PASS

Starting schema test: TEST_UNEXPECTED_TABLES Test that the schema does not contain any unexpected tables

Completed schema test: TEST_UNEXPECTED_TABLES --> Test that the schema does not contain any unexpected tables +++ PASS

Starting schema test: TEST_UNEXPECTED_PROCEDURES Test that the schema does not contain any unexpected stored procedures

Completed schema test: TEST_UNEXPECTED_PROCEDURES --> Test that the schema does not contain any unexpected stored procedures +++ PASS

Starting schema test: TEST_UNEXPECTED_VIEWS Test that the schema does not contain any unexpected views

Completed schema test: TEST_UNEXPECTED_VIEWS --> Test that the schema

```

does not contain any unexpected views +++ PASS
  Starting index test for table MDS_ATTRIBUTES: TEST_UNEXPECTED_INDEXES
--> Test that the table does not contain any unexpected indexes
  Completed index test for table MDS_ATTRIBUTES: TEST_UNEXPECTED_INDEXES
--> Test that the table does not contain any unexpected indexes +++ PASS
  Starting index test for table MDS_LABELS: TEST_UNEXPECTED_INDEXES -->
Test that the table does not contain any unexpected indexes +++ PASS
  Starting index test for table MDS_LARGE_ATTRIBUTES: TEST_UNEXPECTED_IN-
DEXES --> Test that the table does not contain any unexpected indexes
  Starting schema test: TEST_UNEXPECTED_TRIGGERS Test that the schema
does not contain any unexpected triggers
  Completed schema test: TEST_UNEXPECTED_TRIGGERS --> Test that the sche-
ma does not contain any unexpected triggers +++ PASS
  Starting schema test: TEST_UNEXPECTED_COLUMNS Test that tables and
views do not contain any unexpected columns
  Completed schema test: TEST_UNEXPECTED_COLUMNS --> Test that tables and
views do not contain any unexpected columns +++ PASS
  Starting datatype test for table MDS_ATTRIBUTES: TEST_COLUMN_DATA-
TYPES_V2 --> Test that all table columns have the proper datatypes
  Completed datatype test for table MDS_ATTRIBUTES: TEST_COLUMN_DATA-
TYPES_V2 --> Test that all table columns have the proper datatypes +++ PASS
  Starting datatype test for table MDS_COMPONENTS: TEST_COLUMN_DATA-
TYPES_V2 --> Test that all table columns have the proper datatypes
  Starting permissions test: TEST_DBA_TABLE_GRANTS Test that DBA user
has privilege to view all user tables
  Completed permissions test: TEST_DBA_TABLE_GRANTS --> Test that DBA
user has privilege to view all user tables +++ PASS
  Starting schema test: TEST_ENOUGH_TABLESPACE Test that the schema ta-
blespaces automatically extend if full
  Completed schema test: TEST_ENOUGH_TABLESPACE --> Test that the schema
tablespaces automatically extend if full +++ PASS
  Starting schema test: TEST_USER_TABLESPACE_QUOTA Test that tablespace
quota for this user is sufficient to perform the upgrade
  Completed schema test: TEST_USER_TABLESPACE_QUOTA --> Test that table-
space quota for this user is sufficient to perform the upgrade +++ PASS
  Starting schema test: TEST_ONLINE_TABLESPACE Test that schema table-
spaces are online
  Completed schema test: TEST_ONLINE_TABLESPACE --> Test that schema ta-
blespaces are online +++ PASS
  Starting schema test: TEST_DATABASE_VERSION Test that the database
server version number is supported for upgrade
  INFO Database product version: Oracle Database 11g Enterprise Edi-
tion Release 11.2.0.3.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing op-
tions
  Completed schema test: TEST_DATABASE_VERSION --> Test that the database
server version number is supported for upgrade +++ PASS
  Finished readiness check of Oracle Metadata Services with status: FAIL-
URE.

```

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

```

Starting index test for table IAU_COMMON: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes
INFO Audit schema index DYN_EVENT_CATEGORY_INDEX in table IAU_COMMON is
missing the required columns or index itself is missing. This maybe caused by
a known issue, anyway, this missing index will be added in 12.2.2 upgrade.
INFO Audit schema index DYN_EVENT_TYPE_INDEX in table IAU_COMMON is
missing the required columns or index itself is missing. This maybe caused by
a known issue, anyway, this missing index will be added in 12.2.2 upgrade.
INFO Audit schema index DYN_TENANT_INDEX in table IAU_COMMON is missing
the required columns or index itself is missing. This maybe caused by a known
issue, anyway, this missing index will be added in 12.2.2 upgrade.
INFO Audit schema index DYN_USER_INDEX in table IAU_COMMON is missing
the required columns or index itself is missing. This maybe caused by a known
issue, anyway, this missing index will be added in 12.2.2 upgrade.
INFO Audit schema index DYN_COMPONENT_TYPE_INDEX in table IAU_COMMON is
missing the required columns or index itself is missing. This maybe caused by
a known issue, anyway, this missing index will be added in 12.2.2 upgrade.
INFO Audit schema index DYN_USER_TENANT_INDEX in table IAU_COMMON is
missing the required columns or index itself is missing. This maybe caused by
a known issue, anyway, this missing index will be added in 12.2.2 upgrade.
Completed index test for table IAU_COMMON: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes +++ FAIL

```

 **Note:**

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

Understanding the Product Schemas Upgrade

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

The following table describes the basic Upgrade Assistant screens seen in most schema upgrades. Your components may include additional custom screens. These custom screens are documented in your product-specific upgrade documentation.

 **Note:**

The Upgrade Assistant screens that display when you upgrade your schemas vary depending on the options you select and the contents of your pre-upgrade environment. Always use your component-specific upgrade guides to complete an upgrade.

Table 2-4 Upgrade Assistant Screens: Upgrading Product Schemas

Screen Title	Description
Welcome	Provides an overview of the Upgrade Assistant and information about important pre-upgrade tasks.

Table 2-4 (Cont.) Upgrade Assistant Screens: Upgrading Product Schemas

Screen Title	Description
Schemas	Presents a choice of the schema upgrade operation to perform: <ul style="list-style-type: none">• Individually Selected Schemas• All Schemas Used by a Domain
Available Components	When you select Individually Selected Schemas , this 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.
All Schema Component List	When you select All Schemas Used by a Domain , this read-only screen displays all the components and schemas in the specific domain directory that are included in the upgrade.
Prerequisites	Prompts for confirmation that all prerequisites have been met before you continue with the upgrade. Check the boxes before you continue. Note: The Upgrade Assistant does not verify that the prerequisites have been met. For example, the Upgrade Assistant cannot detect whether the servers and processes have been stopped as required.

Table 2-4 (Cont.) Upgrade Assistant Screens: Upgrading Product Schemas

Screen Title	Description
<p>Schema Credentials Screen</p> <p>The screen name changes based on the type of schema selected.</p>	<p>Allows you to enter the information needed to connect to the selected schema and the database that hosts the schema:</p> <ul style="list-style-type: none"> • Database Type The types of databases available in the menu varies depending on the schema you are about to upgrade. The database type chosen for upgrade must be identical to the database type that was selected when the RCU created the schema. If you select Oracle Edition-Based Redefinition (EBR) as the database type, the schema that you are upgrading also must have been created by the RCU using the EBR database type. For example, the Upgrade Assistant never converts schemas from one database type to another. • Database Connect String The location of the database. For example, if you are selecting an Oracle database, you can use a URL in the following format: <i>host:port/db_service_name</i> If you are using a Microsoft SQL Server or IBM DB2 database, select the database type from the drop-down menu and review the text below the field, which provides the syntax required for each database type. Note: The Upgrade Assistant accepts other valid forms of connection strings. For example, the Oracle Database TNS style connection string may also be used. • DBA User Name The database user name used to connect to the database. Note: The DBA user must have sufficient privileges to run the Upgrade Assistant, but the user does not have to have SYS or SYSDBA privileges. See Creating a Non-SYSDBA User to Run the Upgrade Assistant. On certain database platforms user names are case-sensitive, and the DBA user name may include lower case letters. The Upgrade Assistant connects to the name as entered and does not convert it to upper case. • DBA Password The password associated with the specified DBA database user. Click Connect to connect to the database, then select the schema to be upgraded. If the schema that is to be upgraded was created by the RCU in a prior Fusion Middleware release, you see a drop-down menu listing the possible schema names. Note: The component ID or schema name changed for UCSUMS schema as of 12c (12.1.2.0), so the Upgrade Assistant does not automatically recognize the possible schemas and display them in a drop-down list. In the case, you must manually enter the name in the text field. The name can be either <i>prefix_OR-ASDPM</i> or <i>prefix_UMS</i>, depending on the starting point for the upgrade. • Schema Owner The schema user name. For example, DEV11g_MDS. Note: All Oracle Fusion Middleware schema names consist solely of upper case characters on all database platforms. Also, all schema names are stored as upper case in the <i>schema_version_registry</i> table. If you type lower case letters in the Schema User Name field, the Upgrade Assistant converts the name to upper case. • Schema Owner Password The password associated with the specified schema user name. • Edition Name

Table 2-4 (Cont.) Upgrade Assistant Screens: Upgrading Product Schemas

Screen Title	Description
Create Schemas	<p>When Oracle Database enabled for edition-based redefinition is selected as the database type, you must specify the existing edition name.</p> <p>Note: Before upgrading an EBR-enabled schema from Fusion Middleware 11g release or from a previous 12c release, you must first connect to the database server and create an edition on the database server for 12c (12.2.1.3.0). The new edition for 12c (12.2.1.2) must be a child of your 11.1.1.7.0, 11.1.1.9.0, 12.1.2.0, 12.1.3.0, 12.2.1.0, or 12.2.1.1 edition.</p> <p>For more information on creating an edition on the server for edition-based redefinition, see <i>Creating an Edition on the Server for Edition-Based Redefinition</i> in <i>Planning an Upgrade of Oracle Fusion Middleware</i>.</p>
Create Schemas Defaults	<p>This screen appears when the Upgrade Assistant detects that some component schemas may be missing and must be created before the upgrade. The Upgrade Assistant can create these schemas for you using the default tablespace and temporary tablespace settings. To customize these settings run the Repository Creation Utility to create the schemas.</p> <p>Create missing schemas for the specified domain</p> <p>By default this option is enabled. The Upgrade Assistant will attempt to create the missing schemas for the domain using the database connection details and schema owner name provided.</p> <p>If the same password is used for all schemas select Use same passwords for all schemas. Enter and confirm the password in the table. You only have to supply the password once.</p> <p>If you do not want the Upgrade Assistant to create these schemas for you, deselect this option and click Next. You will have to run the Repository Creation Utility to create the schemas.</p>
Create Schemas Progress	<p>When using the Upgrade Assistant to create the missing schemas, the default schema settings are used. If you need to modify the size of the tablespace datafile ,or make any other changes to the default schema settings, use the Repository Creation Utility to create the schemas. You cannot modify the tablespace settings from the Upgrade Assistant.</p> <p>The default datafile size is listed for each component schema and auxiliary schema. If you need to add additional space, use the Repository Creation Utility.</p>
Create Schemas Progress	<p>Displays the status of the schema creation process. No action is required.</p>

Table 2-4 (Cont.) Upgrade Assistant Screens: Upgrading Product Schemas

Screen Title	Description
Examine	<p>Displays the status of the Upgrade Assistant as it examines each component, verifying that the component is ready for upgrade.</p> <p>The Upgrade Assistant examines each component to validate it meets a minimum set of criteria before you begin the upgrade process.</p> <p>If the information is listed in the schema version registry table, this screen displays the Source Version of the schema. If the schema was not created using RCU, or the source version cannot be found, the source version displays <i>unavailable</i>.</p> <p>Status Definitions:</p> <ul style="list-style-type: none"> • <i>in progress</i>: The Upgrade Assistant is examining the upgrade items for the component. • <i>pending</i>: The component will be examined when the Upgrade Assistant finishes the preceding component. • <i>failed</i>: Upgrade items were missing or did not meet the upgrade criteria. The Upgrade Assistant cannot upgrade the component until the issues have been resolved. Click View Log to troubleshoot the errors and then restart the Upgrade Assistant. • <i>succeeded</i>: Upgrade items were found and are valid for upgrade. • <i>canceled</i>: Canceling the examination process has no effect on the schemas or configuration data; the only consequence is that the information that the Upgrade Assistant has collected must be collected again in a future upgrade session. • <i>upgrade not necessary</i>: The component is already at the upgraded version or the component was upgraded in a previous run of the Upgrade Assistant. • <i>skipped</i>: A dependent component has failed resulting in the examine of this component to be skipped. <p>Note: Issues detected during the examination phase may be resolved and the Upgrade Assistant can be started again. However, once the upgrade phase has started, you need to restore your pre-upgrade environment from backup before starting the Upgrade Assistant again.</p>
Upgrade Summary	<p>Displays a summary of the options that you have selected before you start the upgrade process.</p> <ul style="list-style-type: none"> • Expand and collapse the tree to show or hide details about the data provided in the wizard screens, such as schema details and Oracle WebLogic domain directory information. <p>The Summary screen also displays the Source Version of the schema being upgraded and the resulting Target Version post upgrade. Make sure that both versions are correct before proceeding with the upgrade.</p> <ul style="list-style-type: none"> • Click Upgrade to start the upgrade process. <p>If you are upgrading a schema, verify that you have a backup of the database that hosts the schema.</p> <ul style="list-style-type: none"> • Click Save Response File to create a file that can later be used as input to the Upgrade Assistant for a <i>silent upgrade</i>. A silent upgrade performs exactly the same function that the Upgrade Assistant wizard performs, but you do not have to manually enter the data again.
Upgrade Progress	<p>Displays the status of the upgrade process.</p> <p>CAUTION: Do not cancel the upgrade process once it has started. Doing so may leave your components in an inconsistent state requiring a restore from backup.</p>

Table 2-4 (Cont.) Upgrade Assistant Screens: Upgrading Product Schemas

Screen Title	Description
Upgrade Success	Appears when the upgrade is successful. The Post-Upgrade Actions window describes the manual tasks that you must perform to make the component function in the new installation. Note, however, that there may be additional post-upgrade tasks to perform. Consult your upgrade documentation.
Upgrade Failure	<p>Appears when the upgrade fails for the specified component schemas. You must restart the Upgrade Assistant.</p> <p>The Upgrade Assistant logs are available at <code>ORACLE_HOME/oracle_common/upgrade/logs</code></p> <p>Note: If the upgrade fails, you must restore your pre-upgrade environment from back-up, fix the issues, and then restart the Upgrade Assistant. You cannot fix simply the issues and restart the Upgrade Assistant because the files were modified during this operation.</p>

Understanding the Component Configurations Upgrade

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

If you are running the Upgrade Assistant from an Oracle home that contains managed WebLogic domain components, then the **All Configurations Used by a Domain** upgrade option is available.



Note:

The Upgrade Assistant screens that display when you upgrade your component configurations vary depending on the options you select and the contents of your pre-upgrade environment. Always use your component-specific upgrade guides to complete an upgrade.

Table 2-5 Upgrade Assistant Screens: Upgrading Oracle WebLogic Component Configurations

Screen	Description
Welcome	Provides an overview of the Upgrade Assistant and information about important pre-upgrade tasks.
All Configurations Used by a Domain	When the upgrade type selected is All Configurations Used by a Domain , the Upgrade Assistant upgrades component configurations for a managed WebLogic Server domain. Configuration upgrades are performed offline. You must enter the domain directory for the domain that you are upgrading.
WebLogic Server Component List	When the upgrade type selected is All Configurations Used by a Domain , the Upgrade Assistant, this screen provides a list of components that will be included in the WebLogic domain's component configuration upgrade. The name of the domain is provided along with the list of components located within the domain.

Table 2-5 (Cont.) Upgrade Assistant Screens: Upgrading Oracle WebLogic Component Configurations

Screen	Description
Prerequisites	<p>Prompts for confirmation that all prerequisites have been met before you continue with the upgrade. Check the boxes before you continue.</p> <p>Note: The Upgrade Assistant does not verify that the prerequisites have been met. For example, the Upgrade Assistant cannot detect whether the servers and processes have been stopped as required.</p>
Examine	<p>Displays the status of the Upgrade Assistant as it examines each component, verifying that the component is ready for upgrade.</p> <p>The Upgrade Assistant examines each component to validate it meets a minimum set of criteria before you begin the upgrade process.</p> <p>If the information is listed in the schema version registry table, this screen displays the Source Version of the schema. If the schema was not created using RCU, or the source version cannot be found, the source version displays unavailable.</p> <p>Status Definitions:</p> <ul style="list-style-type: none"> • <code>in progress</code>: The Upgrade Assistant is examining the upgrade items for the component. • <code>pending</code>: The component will be examined when the Upgrade Assistant finishes the preceding component. • <code>failed</code>: Upgrade items were missing or did not meet the upgrade criteria. The Upgrade Assistant cannot upgrade the component until the issues have been resolved. Click View Log to troubleshoot the errors and then restart the Upgrade Assistant. • <code>succeeded</code>: Upgrade items were found and are valid for upgrade. • <code>canceled</code>: Canceling the examination process has no effect on the schemas or configuration data; the only consequence is that the information that the Upgrade Assistant has collected must be collected again in a future upgrade session. • <code>upgrade not necessary</code>: The component is already at the upgraded version or the component was upgraded in a previous run of the Upgrade Assistant. • <code>skipped</code>: A dependent component has failed resulting in the examine of this component to be skipped. <p>Note: Issues detected during the examination phase may be resolved and the Upgrade Assistant can be started again. However, once the upgrade phase has started, you need to restore your pre-upgrade environment from backup before starting the Upgrade Assistant again.</p>
Upgrade Summary	<p>Displays a summary of the options that you have selected before you start the upgrade process.</p>
Upgrade Progress	<p>Displays the status of the upgrade process.</p> <p>CAUTION: Do not cancel the upgrade process once it has started. Doing so may leave your components in an inconsistent state requiring a restore from backup.</p>
Upgrade Success	<p>Appears when the upgrade is successful. The Post-Upgrade Actions window describes the manual tasks that you must perform to make the component function in the new installation. Note, however, that there may be additional post-upgrade tasks to perform. Consult your upgrade documentation.</p>

Table 2-5 (Cont.) Upgrade Assistant Screens: Upgrading Oracle WebLogic Component Configurations

Screen	Description
Upgrade Failure	<p>Appears when the upgrade fails for the specified components. You must restart the Upgrade Assistant.</p> <p>The Upgrade Assistant logs are available at <code>ORACLE_HOME/oracle_common/upgrade/logs</code></p> <p>Note: If the upgrade fails, you must restore your pre-upgrade environment from backup, fix the issues, and then restart the Upgrade Assistant. You cannot fix simply the issues and restart the Upgrade Assistant because the files were modified during this operation.</p>

Performing Post-Upgrade Procedures

Complete any additional post-upgrade configuration tasks after the upgrade to verify that your newly upgrade domain is functioning as expected. Perform only those tasks that apply to your domain configuration.

After a successful upgrade it is important to verify that the servers can be started, the schema versions have been updated in the registry table, and the component configurations are correct. In some cases you may need to perform additional post-upgrade tasks based on the contents of your domain. Review the entire list of tasks to determine which are applicable.

Note:

If you are unable to complete one or more of these tasks in your newly upgraded environment, see [Troubleshooting Your Upgrade](#). Always refer to your component-specific upgrade documentation for more information on post-upgrade procedures.

- [Performing Basic Post-Upgrade Administrative Tasks](#)
Review the list of post-upgrade tasks and perform those that apply to your upgraded environment and domain configurations.
- [Verifying a Successful Schema Upgrade](#)
- [Checking for Invalid Database Objects After an Upgrade](#)
If you are using an Oracle database, recompile database objects after running the Upgrade Assistant.

Performing Basic Post-Upgrade Administrative Tasks

Review the list of post-upgrade tasks and perform those that apply to your upgraded environment and domain configurations.

These administrative tasks are optional, but Oracle strongly recommends that you verify your upgrade by performing the tasks.

Table 2-6 Basic Post-Upgrade Administration Tasks

Task	Description	More Information
Starting and stopping products and servers	Start and stop Oracle Fusion Middleware, including the Administration Server, Managed Servers, and components. Performing these tasks validates that the upgrade was successful.	Starting and Stopping Oracle Fusion Middleware
Starting and stopping upgraded applications	Start your upgraded applications in the new environment to verify they are working as expected.	Starting and Stopping Applications
Configuring Secure Sockets Layer (SSL)	Set up secure communications among between Oracle Fusion Middleware components using SSL.	Configuring SSL in Oracle Fusion Middleware
Deploying Applications	Deploy your applications to Oracle Fusion Middleware.	Deploying Applications
Monitoring Oracle Fusion Middleware	Keep track of the status of Oracle Fusion Middleware components.	Monitoring Oracle Fusion Middleware
Adding a Web Tier front-end to your WebLogic domain	Oracle Web Tier hosts Web pages (static and dynamic), provides security and high performance along with built-in clustering, load balancing, and failover features. In particular, the Web Tier contains Oracle HTTP Server.	<i>Installing and Configuring Oracle HTTP Server</i>
Tuning and configuring Coherence for your topology	The standard installation topology includes a Coherence cluster that contains storage-enabled Managed Coherence Servers. This configuration is a good starting point for using Coherence, but depending upon your specific requirements, consider tuning and reconfiguring Coherence to improve performance in a production environment.	For information about Coherence clusters, see <i>Configuring and Managing Coherence Clusters in Administering Clusters for Oracle WebLogic Server</i> . For information about tuning Coherence, see <i>Administering Oracle Coherence</i> . For information about storing HTTP session data in Coherence, see <i>Using Coherence*Web with WebLogic Server in Administering HTTP Session Management with Oracle Coherence*Web</i> . For more information about creating and deploying Coherence applications, see <i>Developing Oracle Coherence Applications for Oracle WebLogic Server</i> .

Verifying a Successful Schema Upgrade

You can use the following SQL command to verify that the schema version in `schema_version_registry` has been properly upgraded:

```
SET LINE 120
COLUMN MRC_NAME FORMAT A14
COLUMN COMP_ID FORMAT A20
COLUMN VERSION FORMAT A12
COLUMN STATUS FORMAT A9
COLUMN UPGRADED FORMAT A8
SELECT MRC_NAME, COMP_ID, OWNER,
VERSION, STATUS, UPGRADED FROM
SCHEMA_VERSION_REGISTRY ORDER BY MRC_NAME, COMP_ID ;
```

Make sure that the numbers in the `VERSION` column are updated. If no changes were required, some schemas may remain at their pre-upgrade version. For more information, see [Table 1-1](#).

In the query results, the `STATUS` field is either `UPGRADING` or `UPGRADED` during the schema patching operation, and becomes `VALID` when the operation is finished.

If the status appears as `INVALID`, then the schema upgrade has failed. You can examine the log files to determine the reason for the failure.

Checking for Invalid Database Objects After an Upgrade

If you are using an Oracle database, recompile database objects after running the Upgrade Assistant.

To determine whether any database objects were corrupted during the upgrade, recompile the database objects that were upgraded by Upgrade Assistant by connecting to the database as `SYS` and running the following from SQL*Plus:

```
SQL> @oracle_home/software/rdbms/admin/utlrbp.sql
```

Enter the following query to ensure there are no longer any invalid database objects:

```
SELECT owner, object_name FROM all_objects WHERE status='INVALID' ;
```

None of the database objects for the upgraded schema should be invalid at this point. If there are any, run the `utlrbp.sql` command again and check again. If the problem persists, you should file a service request.

3

Performing a Silent Upgrade with a Response File

A silent upgrade is an upgrade that requires no user interaction. You can perform a silent upgrade by using a response file.

A response file collects all the information that you enter through the Upgrade Assistant's graphical user interface (GUI) screens. Later, you can use the response file to automatically perform exactly the same function that the Upgrade Assistant wizard performs.

Note:

You must run the Upgrade Assistant in GUI mode first to generate a response file that can be used to complete a silent upgrade.

- [Creating an Upgrade Response File](#)
When you run the Upgrade Assistant in graphical user interface (GUI) mode, you can select to generate a response file.
- [Using a Response File to Upgrade Fusion Middleware](#)
When you create a response file to record the information that you enter through the Upgrade Assistant's screens, you can perform the same actions later by using the command line in *silent mode*.

Creating an Upgrade Response File

When you run the Upgrade Assistant in graphical user interface (GUI) mode, you can select to generate a response file.

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

When you select the **Save Response File** option, you are prompted for a name and location where you want to save the response file. After it is created, you can use it as isto replicate the upgrade options on other systems, or modify it as needed.

Using a Response File to Upgrade Fusion Middleware

When you create a response file to record the information that you enter through the Upgrade Assistant's screens, you can perform the same actions later by using the command line in *silent mode*.

Note:

You must run the Upgrade Assistant in GUI mode first to generate the response file that can be used to complete a silent readiness check or upgrade.

To start the Upgrade Assistant using a response file:

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

2. Enter the following command:

```
(UNIX) ./ua -response response_file[-examine] [-logLevel log_level] [-logDir log_directory] [-threads number]
```

```
(Windows) ua.bat -response response_file[-examine] [-logLevel log_level] [-logDir log_directory] [-threads number]
```

For information about the optional command-line parameters, see:

- [Upgrade Assistant Parameters](#)

Upgrade Assistant Parameters

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

Table 3-1 Upgrade Assistant Command-Line Parameters

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

Table 3-1 (Cont.) Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
-threads	Optional	Identifies the number of threads available for concurrent schema upgrades or readiness checks of the schemas. The value must be a positive integer in the range 1 to 8. The default is 4.
-response	Required for silent upgrades or silent readiness checks	Runs the Upgrade Assistant using inputs saved to a response file generated from the data that is entered when the Upgrade Assistant is run in GUI mode. Using this parameter runs the Upgrade Assistant in <i>silent mode</i> (without displaying Upgrade Assistant screens).
-examine	Optional	Performs the examine phase but does not perform an actual upgrade. Do not specify this parameter if you have specified the <code>-readiness</code> parameter.
-logLevel <i>attribute</i>	Optional	Sets the logging level, specifying one of the following attributes: <ul style="list-style-type: none">• TRACE• NOTIFICATION• WARNING• ERROR• INCIDENT_ERROR The default logging level is NOTIFICATION. Consider setting the <code>-logLevel TRACE</code> attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if <code>-logLevel TRACE</code> is used.

Table 3-1 (Cont.) Upgrade Assistant Command-Line Parameters

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

4

Troubleshooting Your Upgrade

Before contacting Oracle Support, you can troubleshoot common problems and issues that may occur while you are using the Upgrade Assistant to upgrade your Oracle Fusion Middleware deployment to this release.

- [Troubleshooting an Upgrade](#)
Understanding the errors that you may encounter while running the Upgrade Assistant will help you to effectively troubleshoot them.
- [Reviewing Log Files](#)
Do not delete the log files. They help diagnose and correct the problem while you run the Upgrade Assistant.
- [Investigating Examination Failures](#)
Identify and troubleshoot failures that occur during the Examination phase of the upgrade.
- [Investigating Upgrade Failures](#)
Identify and troubleshoot failures that occur during the upgrade.
- [Resolving Common Upgrade Assistant Errors](#)
Attempt to resolve common Upgrade Assistant errors before contacting Oracle Support.
- [Attempting to Upgrade an Unsupported Domain](#)
Do not attempt to upgrade the schemas or domain configurations in an unsupported domain.
- [Restarting the Upgrade Assistant After a Failure](#)
You must resolve errors before you restart the Upgrade Assistant.

Troubleshooting an Upgrade

Understanding the errors that you may encounter while running the Upgrade Assistant will help you to effectively troubleshoot them.

 **Note:**

Your course of action depends on the phase in which the error occurred.

If errors occur while you are running the Upgrade Assistant, use the following steps to troubleshoot the problem:

1. Locate and open the Upgrade Assistant log file with a text editor:
For information about the location of the log file, see [Reviewing Log Files](#).
2. To verify the correct version of the Upgrade Assistant .jar file:
 - a. Go to the the `jlib` directory:

```
cd ORACLE_HOME/oracle_common/upgrade/jlib
```

- b. Enter the following command:

```
unzip -p ua.jar META-INF/MANIFEST.MF
```

 **Note:**

The output of the `unzip -p ua.jar META-INF/MANIFEST.MF` command identifies the development label that was used to build the Upgrade Assistant, and that information identifies the date and version of the Upgrade Assistant software that was run. If you submit a service request, you need to provide this information to Oracle.

3. Locate any error messages that are identified by number; for example, UP-GAST-00091.
4. Look up the error number in *Error Messages*.

The error description may include the cause of the error, as well as the action you should take to resolve the error.

5. Based on whether you can locate an error message and the error message description, do one of the following:
 - If you are able to identify a solution to the upgrade failure, implement your solution, then restart the Upgrade Assistant to perform the upgrade again.

When you rerun the Upgrade Assistant, any components that were upgraded successfully during the previous run are not affected. However, the Upgrade Assistant attempts to upgrade any components that were not upgraded successfully.
 - Contact Oracle Support about any errors that are not documented or that cannot be resolved by following documented actions. Note that some errors that occur require the repository to be restored from backup, the problem to be resolved, and another upgrade to be run. Note that Oracle Support requires you to provide both the `UA.log` file and the `UA.out` file, if present. Provide complete logs and not just excerpts of those files.

 **Note:**

If you get any errors during Examination phase, and no components or schemas have been upgraded yet, run the readiness check. The types of checks performed by the readiness check are more thorough than Examination phase of upgrade.

Errors that occur during or after the upgrade, however, require you to restore your environment from your backup copies, correct the errors and then restart the upgrade process from the beginning.

Reviewing Log Files

Do not delete the log files. They help diagnose and correct the problem while you run the Upgrade Assistant.

When running the Upgrade Assistant, you can alter the contents of your log files by specifying a different `-logLevel` value on the command line. The default value is `-logLevel NOTIFICATION`. You can alter the location of your log files using the `-logDir` parameter. You can obtain more detailed logging information by running the Upgrade Assistant with `-logLevel TRACE`. However, this may cause the log file to become very large.

 **Note:**

`TRACE` messages are not included in the Upgrade Assistant Log File Viewer. To view `TRACE` messages, use another tool.

 **Tip:**

To expedite the review process, search for the word `ERROR`.

For more information about the error messages in your log files, see [Resolving Common Upgrade Assistant Errors](#).

Log files are stored in the following default directory:

- (UNIX) `ORACLE_HOME/oracle_common/upgrade/logs/uatimestamp.log`
- (Windows)

`ORACLE_HOME\oracle_common\upgrade\logs\uatimestamp.log`

where `timestamp` states the actual date and time when the Upgrade Assistant was run.

Some components create a second log file called `uatimestamp.out`, also in the same location.

For database schema upgrades of certain components, there can also be an output (`.out`) file that contains the screen output of commands that were run in a shell process or as PL/SQL scripts. You can locate these output files in the same default directory.

If there are questions or issues about an upgrade failure that cannot be resolved with the information in this guide, it is important to retain the log files. If you submit a service request, you must upload the entire Upgrade Assistant `.log` file and the `.out` files (if available) along with the service request.

Investigating Examination Failures

Identify and troubleshoot failures that occur during the Examination phase of the upgrade.

To determine the cause of an Examination failure:

1. Note the name of the failed component in the Upgrade Assistant dialog or command-line output.
2. Open the following Upgrade Assistant log file.

For information about the location of the log file, see [Reviewing Log Files](#).

3. In the log file, search for the message `Starting to examine component_name`.

To complete the upgrade, resolve the issues and then launch the Upgrade Assistant again, or, if possible, click **Back** to return to a previous screen and make the necessary changes.



Note:

The readiness check performs checks to a far greater level of detail than the Examination phase. If `Examine` fails, you can run the Upgrade Assistant with the `-readiness` parameter and make sure that the report does not show any test failures.

Issues detected during the Examination phase can be resolved without restoring from backup (as is required to resolve errors encountered during the actual upgrade). However, if you attempt to resolve an Examination error in a way that changes the state of the system, you need to restore the entire system to the pre-upgrade state (before any upgrade operations were attempted).

Investigating Upgrade Failures

Identify and troubleshoot failures that occur during the upgrade.

To determine the cause of an upgrade failure:

1. Note the name of the failed component in the Upgrade Assistant dialog or command-line output.
2. Open the upgrade log file:

For information about the location of the log file, see [Reviewing Log Files](#).

3. Search for the message `Starting to upgrade component_name`.

To complete the upgrade, restore the entire environment using your pre-upgrade backup to a point in time before any upgrade operations were attempted, resolve the issues, then launch the Upgrade Assistant again. You will have to start the upgrade process from the beginning to ensure a successful (complete) upgrade.

 **Note:**

You should back up all databases with `RMAN` and be able to do a point-in-time recovery from those backups. If the Fusion Middleware repository for your domains spans multiple Oracle Database servers, you must restore from each of those backups.

Resolving Common Upgrade Assistant Errors

Attempt to resolve common Upgrade Assistant errors before contacting Oracle Support.

The following sections provide descriptions of the most common upgrade errors. For a complete list of Fusion Middleware errors, see *Error Messages*.

- [Ensuring there is sufficient disk space](#)
- [Resolving Database Connection Problems When Upgrading Schemas](#)

Ensuring there is sufficient disk space

If an upgrade fails because the database server has run out of disk space, you must restore the database server environment from backups, add sufficient disk space or remove unwanted files (such as temp or trace files) from the database server, and then retry the upgrade.

 **Note:**

Once a database schema upgrade has failed due to this class of error, you cannot simply add more disk space and retry the upgrade. The schemas have been left in an inconsistent state and may have been marked `INVALID`. You cannot recover from this error without restoring the original database state from backups.

The following examples show some insufficient disk space errors that you may encounter:

ORA-01658: unable to create INITIAL extent for segment in tablespace

Cause: The existing schema tablespace does not have sufficient space to complete the upgrade.

Action: Make sure that the tablespace has sufficient room (space) for a successful upgrade. Oracle recommends that you add more data files to the existing database tablespaces, otherwise the upgrade will fail.

ORA-01114: IO error writing block to file *block number*

Cause: The device on which the file resides is probably offline. If the file is a temporary file, then it is also possible that the device has run out of space. This could happen because disk space of temporary files is not necessarily allocated at file creation time.

Action: Restore access to the device or remove unnecessary files to free up space.

ORA-09945: Unable to initialize the audit trail file

Cause: The system is unable to write header information to the file being used as the audit trail. The `audit_trail_dest` or audit trail destination is full for generation of audit file.

Action: Free up space and retry the operation.

Resolving Database Connection Problems When Upgrading Schemas

If you have trouble connecting to a database when using the Upgrade Assistant to upgrade a component schema, try connecting to the database using another tool, such as SQL*Plus. This helps you troubleshoot the problem by verifying that the database is up and running and available on the network.

Attempting to Upgrade an Unsupported Domain

Do not attempt to upgrade the schemas or domain configurations in an unsupported domain.

If you receive an error stating that the specified domain cannot be upgraded, you must first upgrade the domain to a supported version. *Do not attempt to upgrade schemas or domain configurations in an unsupported domain.*

Restarting the Upgrade Assistant After a Failure

You must resolve errors before you restart the Upgrade Assistant.

If the Upgrade Assistant fails during the upgrade phase, or only partially upgrades your components, try to resolve the issues and then follow these steps:

1. Identify and resolve the issue. Review the log files and note that you may need to apply a patch.

If you continue to experience upgrade failures, consider setting `-logLevel` to `TRACE` so that more information is logged. This is useful when troubleshooting a failed upgrade, but be sure to reset `-logLevel` to `NOTIFICATION` after the issue has been resolved to avoid performance issues.

2. Restore the entire 11g or 12c environment from your pre-upgrade backup.
3. Restart the upgrade as described in your component-specific upgrade guide.

If you cannot follow the standard backup and recovery procedures recommended by Oracle, and you must perform an online restoration.